CO-SIMULATION AND DESIGN OF CYBER-PHYSICALLY SECURE BULK ELECTRIC POWER GRIDS

A Dissertation
Presented to
The Academic Faculty

by

Victor Chukwuka

In Partial Fulfillment
Of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology
December 2018

CO-SIMULATION AND DESIGN OF CYBER-PHYSICALLY SECURE BULK ELECTRIC POWER GRIDS

Approved by:

Professor Santiago Grijalva, Advisor
School of Electrical & Computer Engineering
Georgia Institute of Technology

Professor Raheem Beyah
School of Electrical & Computer Engineering
Georgia Institute of Technology

Professor Maryam Saeedifard
School of Electrical & Computer Engineering
Georgia Institute of Technology

Professor Thomas Habetler
School of Electrical & Computer Engineering
Georgia Institute of Technology

Professor Masoud Nazari
School of Electrical Engineering
California State University

Date Approved: August 16, 2018

ACKNOWLEDGMENTS

I would like to thank each committee member for their assistance and valuable unique feedback on this dissertation. I am grateful to my advisor, Prof. Santiago Grijalva, for giving me the opportunity to join the Advanced Computational Electricity Systems (ACES) Lab, where I got to work on problems I was both passionate and excited about. I would like to thank Prof. Raheem Beyah, and Prof. Masoud Nazari for the many discussions, suggestions, and helpful comments on my research work.

I am grateful to have been a part of the many discussions I had with friends and colleagues in the ACES laboratory during my time at Georgia Tech. I would like to thank Dr. Masoud Nazari for the many long hours spent attacking electric grid wireless communication problems from different perspectives logically and methodologically. To Dr. Neetesh Saxena, my thanks for being a great research colleague, roommate, and friend. And to the other lab members, Leilei Xiong, and Jeremiah Deboever, who made my time in the ACES lab one to remember, I say thank you! I would also like to thank the professors who gave me an opportunity to perform research under their supervision and helped me gain confidence as a student researcher. To Prof. Jeffrey Shield at the University of Nebraska-Lincoln and Prof. Ilesanmi Adesida at the University of Illinois, Urbana-Champaign, thank you for your tutelage and probing research questions that helped inch me closer towards being a perseverant researcher.

I would like to thank the friends I made during my time in graduate school who impacted my journey in one way or another. I would like to thank Ogundiran Soumonni, Keerthi Suria Kumar Arumugam, Chukwuemeka Obikwelu, Dickson Nosegbe, Chu Meh Chu, Phillip Asare, Hadi Afrasiabi, Mehdi Nikkhah, Nikolay Atanasov, and Arman Khouzani.

Outside of Georgia Tech, I would like to acknowledge a few people who were pillars of support and sources of encouragement in completing this journey. To Keldreca Thomas, for always being my calm in the storm, you are the real 'MVP,' and I appreciate you! Thanks to Ninrat Datiri, for always encouraging me to strive to be the best me I can be, and to Maja Niksic, for being a dear friend and accountability partner, and always holding me to the highest standard there is. I would like to thank Zegbeh Kpadeh for being my close friend and confidant when I first arrived in the US. We challenged ourselves to seek knowledge, and we did so at the highest level!

I would also like to thank several administrative staff and personnel at the Georgia Institute of Technology who have helped me with different issues during my time here. I would like to thank Etta Pittman, Tasha Torrence, Jacqueline Trappier, Christopher Malbrue, Daniela Staiculescu, and Felicia Benton-Johnson with the Center for Engineering Education and Diversity (CEED), and Terrance Gresham with the Scheller School of Business.

Above all, and most importantly, to my family, I am deeply indebted to you all for your support and encouragement over the years. Many thanks go to my father, Christopher Chukwuka, for his words of wisdom and instilling the importance of education in my brothers and me. To my mother, Grace Chukwuka, who is always praying for me, and from whom I got my tenacious nature, I say thank you. I would like to give special thanks to my brothers, Jeremiah Chukwuka, for always urging me to go out of my comfort zone, Abraham Chukwuka, for always being my voice of reason, and Joshua Chukwuka, for being my personal prayer warrior. Thank you all for encouraging and believing in me. We did it!!!

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

The control of Bulk Electric Systems (BES) uses communications, software, and embedded systems, which makes BES vulnerable to cyber-physical security attacks. Cyber-Physical security attacks occur through cyber elements with the intention to disrupt physical system operations or physically damage power devices. Cyber-physical attack mechanisms include targeting the communications bandwidth or injecting bad data, which may result in bad data being injected into the power delivery system or a component being tricked into executing a seemingly benign command that harms the overall system or limits the number of messages that can reach their destination.

The objective of this dissertation is to 1) develop algorithms that capture how bad data injection attacks propagate in a power delivery system, 2) develop a tool that can model a bad command injection in bulk electric systems, and 3) develop a cyber-physical method and metric for quantifying the effect of a cyber-physical attack on bulk electric systems. We begin with a discussion of motivating the shift from traditional IT cyber-security to the new paradigm of cyber-physical security and describe its characteristics. By using power system state estimation, we develop a graph-based attack propagation model that simulates a bad data injection attack and executes a heuristic defense strategy. Next, we develop a co-simulator that models and simulates both the power system and the communication in an integrated manner. This provides a capability for analyzing the overall cyber-physical security of the entire system by 1) characterizing system behavior under different attack scenarios, and 2) quantifying system cyber-physical security through cyber-physical security assessment (CPSA) metrics that provide insight into impact analysis of cyber-physical attacks on the system. We develop an attack model and a co-simulation framework for simulating the effects of a bad command injection on two bulk electric system test

cases. We also develop an enhanced visualization prototype for increased operator situational awareness of the cyber-physical security status of the BES. The results indicate that modeling and simulation (M&S) of cyber-physical security attacks holds promise as a way of studying and understanding how cyber-physical security attacks in bulk electric systems affect the system components and suggest the need for/the benefit of the implementation of cyber-physical security assessment modules into existing control systems to manage such real-world attacks when they occur on BES.

CHAPTER I

INTRODUCTION

1.1. Background

The electrical grid is the largest, most complex machine ever created. As a result of continuing rapid advancements in information, computing and communication technologies, it is currently evolving into the "smart grid." As shown in Figure 1, the smart grid makes use of ubiquitous sensors, high-speed communication networks, and Internet of Things (IoT) connectivity to both facilitate grid functions and enable the integration of distributed renewable energy sources and energy storage into the grid while ensuring its overall efficiency, reliability, and cyber-physical security. Managing and coordinating such a complex system involves huge information and data flow across the system. Key technologies such as the supervisory control and data acquisition (SCADA) and distributed control systems (DCS) provide visibility into and enable efficient and intelligent control of grid functions, assets, and resources.

The traditional electric grid, which is divided into generation, transmission, and distribution domains, is hierarchical in nature. In the emerging smart grid, distributed energy resources (DER) such as wind, solar, electric vehicles, demand response and energy storage are transforming traditional consumers into *prosumers*, who, in addition to consuming electricity, can generate their own energy and sell it back to the grid. Prosumers range from residential to commercial buildings and larger microgrids. SCADA, DCS, remote terminal units (RTUs), merging units, intelligent electronic devices (IEDs), and phase measurement units (PMUs) in substations throughout the grid poll and collate raw field measurements that are then used to estimate the current state of the grid through state estimation performed at the control center [1]. The energy management system

(EMS), in coordination with the control centers, uses information garnered from these sensors to send command and control signals to intelligent electronic devices (IES) and corresponding actuation functions to ensure that power generation always matches power demand so that voltages are within operational ranges and that, in general, the physical operation of the grid is secure.

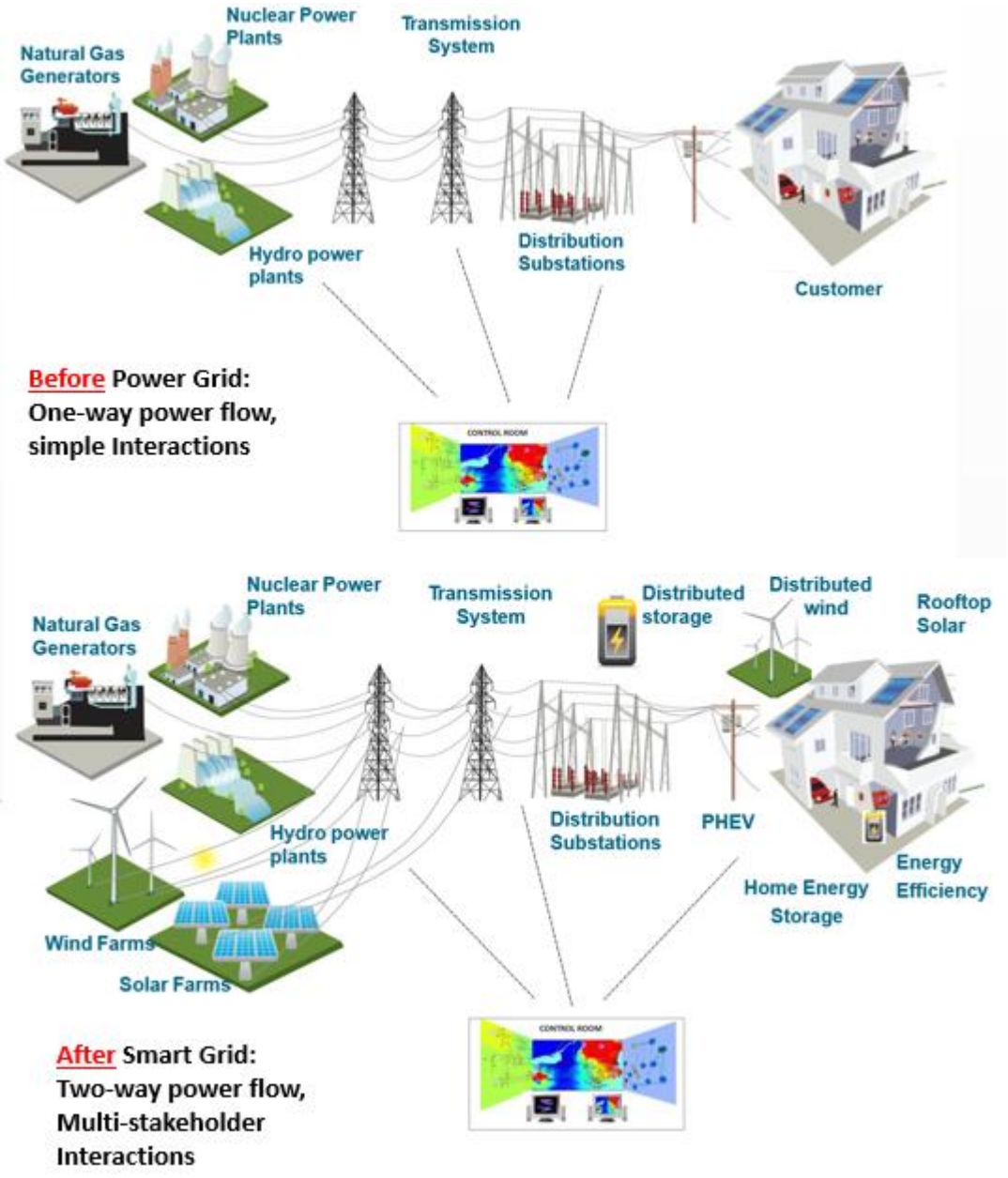

Figure 1 - Traditional Grid vs. Smart Grid

While conventional power systems are well established, investigating the cyber-physical security challenges faced by the smart grid due to advancements in IoT technologies requires first understanding what the emerging smart grid actually is. Title XIII of the Energy Independence and Security Act of 2007 highlights ten characteristics of a Smart Grid:

(1) Increased use of digital information and controls technology to improve reliability, security, and efficiency of the electric grid.

(2) Dynamic optimization of grid operations and resources, with full cybersecurity.

(3) Deployment and integration of distributed resources and generation, including renewable resources.

(4) Development and incorporation of demand response, demand-side resources, and energy-efficiency resources.

(5) Deployment of "smart" technologies (real-time, automated, interactive technologies that optimize the physical operation of appliances and consumer devices) for metering, communications concerning grid operations, grid status, and distribution automation.

(6) Integration of "smart" appliances and consumer devices.

(7) Deployment and integration of advanced electricity storage and peak-shaving technologies, including plug-in electric and hybrid electric vehicles, and thermal storage air conditioning.

(8) Provision to consumers of timely information and control options.

(9) Development of standards for communication and interoperability of appliances and equipment connected to the electric grid, including the infrastructure serving the grid.

(10) Identification and lowering of unreasonable or unnecessary barriers to adoption of smart grid technologies, practices, and services.

The capabilities highlighted above are enabled by new technologies and components that can become targets of a cyber-physical security attack, e.g., attacks that attempt to disrupt the physical devices and physical functionality of the power delivery system. A further challenge of securing the electric grid is that the energy delivery system is vital and must always continue to deliver energy to end-users.

1.2. Context and Motivation

As a result of technological and socioeconomic changes, societies increasingly rely on electricity systems to support most of their activities. More devices are being connected to the grid at all levels: distributed solar generation to augment traditional generation, IEDs in substations to facilitate faster and more efficient delivery of electricity, advanced metering infrastructure (AMI) meters that give users insight and control of their consumption in home area networks (HANs), electric vehicles, and so on. The increased number of connected devices increases the threat surface and makes the grid more vulnerable to cyber hackers and insiders looking to compromise outdated and poorly protected cyber components. Following are key considerations that have led to cyber-physical security challenges that the smart grid is facing today:

- The electrical grid was not built with cyber or cyber-physical security in mind. It was built to be reliable with the use of industrial control systems (ICS) that are now decades old and cannot be easily updated [2]. In addition, the grid's evolution has been a fusion of bolt-on, patching, and buildouts. This presents challenging security problems as the grid evolves from a vertically integrated electric utility model—where one company controls the generation, transmission and distribution—to a decentralized architecture with multiple players [3].

- Multiple players are now involved since power generation has to come from both traditional power plants as well as from distributed generation sources simultaneously. However, the cyber devices deployed by these multiple players are manufactured by a variety of vendors, who implement different communications and security protocols, which creates in the smart grid challenging interoperability problems. It is these interoperability problems that can create more ways for hackers to access and compromise the smart grid.
- One way to mitigate the effects of a cyber-attack on the smart grid is to use distributed generation, control, and automation. However, this approach also increases the threat surface area, since distributed hubs for power generation and control are not as secure as centralized control and data centers.
- Obtaining higher resolution measurements in the new smart grid requires SCADA systems to perform more coordination and control of the different players in the grid. These systems are currently implemented over wide-area IP networks since many of the key players and prosumers are geographically dispersed. As a result of these challenges, malevolent actors in cyberspace have more access points through which they can illegitimately access and compromise the smart grid.

In order to properly capture cyber-security aspects in the smart grid, security models have to consider not just the cyber threats and risks affecting the communications network but also the physical impacts on the power system. Questions regarding how cyber-attacks affect the operations of these devices have to be answered. The interactions between the cyber layer and the physical layer give rise to the domain of cyber-physical security. The grid is increasingly becoming more complex; however, our understanding of its complexities, vulnerabilities, and security dynamics is far from complete. Currently, traditional cybersecurity measures are used as preventive

countermeasures for the smart grid, which is a cyber-physical system and more than just a communication network. However, these traditional cyber-security measures have proved either insufficient, inapplicable, or simply ineffective in preventing, detecting, or mitigating cyber-physical security attacks. Hence, new models, as well as revisions to existing practices must be created to provide better countermeasures to meet the cyber-physical security challenges facing the smart grid today.

Recent cyber-physical attacks on electrical grids in various countries have reinforced the notion that electric grids are targets of a major cyber attack. To put this into context and help highlight the vulnerabilities of the smart grid, Table 1 below presents some of the well-known attacks on the industrial control systems:

Table 1 - Major Cyber Security Attacks on the Energy Sector

	Aurora	**Stuxnet**	**Dragonfly/Havex**	**Sandworm**	**Ukrainian Electric Utility**
Year	2007	2010	2013	2014	2015
Scope	Simulated Attack by Idaho National Labs to Department of Homeland Security showing what happens when a generator is remotely controlled by hackers	Physical Damage to Iran's Uranium enrichment equipment in a nuclear facility	Group that attempts to compromise the cybersecurity of energy companies	Spear-phishing attack targeting SCADA systems and exploiting Windows operating system vulnerability to deliver variants of the BlackEnergy Trojan	3 Power Distribution substations taken offline by hijacking SCADA equipment and HMIs

Table 1 continued

Duration of Attack	-	10 months	2013 – Present	-	Several hours
Number of Customers Affected	-	-		-	225,000
Damage ($)	-	$5 – 10M		-	~ $10M

The electric grid is core to the functioning of U.S. society. The other sixteen critical infrastructures that make up the economy rely heavily on electricity to function properly. The potential consequences of a disruptive, destabilizing, or incapacitating cyber attack on the U.S. smart grid have motivated the research community, the government, and the energy industry to investigate U.S. power grid vulnerabilities and then use the findings to determine and deploy mechanisms to minimize the risk of cyber-attacks.

1.3. Research Objective

The complexity of smart grid cyber-physical security, the interactions between the power systems and the communications networks, and the computational cost to simulate and assess smart grid cyber-physical security attacks call for effective and efficient co-simulation tools. These tools will be critical in providing support for operator decisions and improvement in cyber-attack defense strategy. Given the above considerations, below are the major objectives for the research in this thesis:

1. Design and implement an attack generation framework for evaluating the impact of cyber-physical security attacks.

2. Characterize system behavior under different attack scenarios.
3. Implement a flexible and modular design for a cyber-physical security assessment software tool.
4. Characterize and quantify system cyber-physical security through cyber-physical security assessment (CPSA) metrics that provide insight into impact analysis of cyber-physical attacks on the system.
5. Develop a visualization prototype that captures real-time system dynamics of cyber-physical security.

Because any testing performed on real power systems would be difficult, expensive, and unsafe, the key methodology for studying CPSA will be simulation and analysis. These simulations, which will be performed with realistic data, will allow the study of how different attacks can be constructed and which assets, if taken down, could have the most impact on the system. Because of the inherent complexity in modeling a cyber-physical system by tying the communication and power layer together, we must first introduce a few assumptions and limit the scope of the problem to be investigated. Below s are the assumptions employed in the design and implementation of the CPSA tool:

1. The evaluation of smart grid cyber-physical security will start from already conducted research on constructing undetectable cyber-physical attack models through full or partial knowledge of the power system topology with necessary modifications.
2. Specifically, the approach to the research in this dissertation assumes that the smart grid can be modeled as a complex network to which current topological vulnerability metrics from complex network studies can be applied.

3. The power flow, state estimation, and contingency analysis that are widely used in traditional power grids are central to the computational module in the developed CPSA tool.
4. Because it would be computationally inefficient to replicate a large-scale power system with tens of thousands of components with the associated analogs, status points, and controls, we adopt a model that simplifies several complex real-time dynamics.

The rest of the thesis is organized as follows: Chapter 2 surveys the literature of the bulk electric system vulnerabilities and state-of-the-art approaches for securing such systems against a cyber-physical attack. Chapter 3 presents our approach to modeling cyber-physical security in bulk electric systems, and Chapter 4 presents an attack-graph propagation model for bad data injection into power delivery systems. Chapter 5 presents the modular design of the Cyber-physical Security Assessment (CPSA) co-simulator developed in this dissertation. Chapter 6 describes the co-simulation software framework implementation details, while Chapter 7 presents the cyber-physical security assessment of different test cases using the CPSA tool. Chapter 8 provides conclusions regarding this research and directions for future work.

CHAPTER II

LITERATURE REVIEW

This chapter presents a literature survey of several topics relevant to this dissertation. Section 2.1 gives a description of the Bulk Electric System (BES) and its constituent interconnections and entities. The control and monitoring of the bulk electric system is discussed in section 2.2. Section 2.3 presents the cyber-physical security threats and vulnerabilities facing bulk electricity delivery. Historical approaches and current trends of securing bulk electric systems are discussed in section 2.4, while section 2.5 details existing research work in cyber-physical security modeling and simulation and the challenges that exist. This chapter concludes by describing the need for bulk electric system cyber-physical security co-simulation.

2.1. Bulk Electric System

The bulk electric system, also known as the power transmission system, is the network of interconnected generation and transmission lines designed to transport large amounts of electricity over long distances by using high transmission voltages of 110kV and above. The North American power network is subdivided into five asynchronous regions, as shown in Figure 2. The five synchronous regions are the Eastern and Central United States (including Ontario, Nova Scotia, New Brunswick, Manitoba and Saskatchewan of Canada), Quebec and Labrador, the Western United States and parts of Western Canada including parts of north-western Mexico, and the eastern interconnection and Electric Reliability Council of Texas (ERCOT) in Texas and the rest of Mexico [4].

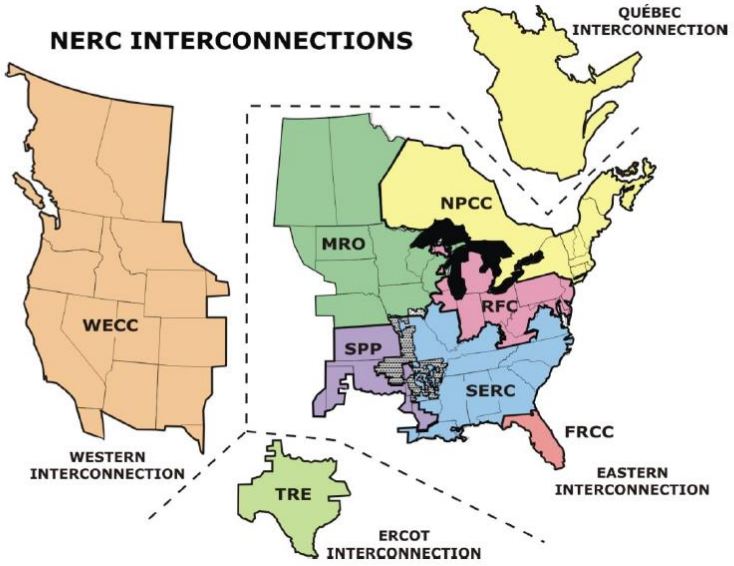

Figure 2 - NERC Synchronous Interconnections and Regional Entities

When the bulk electric system was first designed decades ago, it was fitted with minimal protection using line switches which were sometimes manual. The bulk electric system has evolved over the decades. Today, it is equipped with sensors for transmitting information used for protection, control, monitoring, and security purposes. At the core of bulk electric system are Supervisory control and data acquisition systems (SCADA) and energy management systems (EMS).

Cyber-physical security attacks targeting a BES can be designed to affect two main domains: the monitoring and data gathering capabilities of SCADA, such as a distributed denial of service attack on a remote terminal unit thus preventing it from sending data back to the control center, or the computation capabilities of an EMS, which can be compromised through bad data injection attacks on the state estimator, which can cause the EMS to output incorrect states of the system.

2.2. Control and Monitoring of Bulk Electric System

The control and monitoring of BES is done via two main systems: supervisory control and data acquisition (SCADA) and energy management systems (EMS). SCADA is responsible for gathering and sanitizing field measurements, which are then fed into the EMS to perform a variety of important grid functions such as power flows and optimal dispatch.

2.2.1. SCADA Systems

SCADA is a type of industrial control system (ICS). Industrial control systems are a combination of several types of control systems and instrumentation that monitor and control industrial processes that exist in the physical world. SCADA systems are typically different from other ICSs because they are large-scale processes that operate across multiple geographic sites. The IEEE Std C37.1-1994 [5] defines the SCADA system as a system for remote monitoring and control that operates with coded signals over communication channels to acquire information about remote equipment for monitoring and control functions.

SCADA control system architecture provides a means for remotely monitoring and controlling industrial systems via data communication between computers. The control system is also equipped with human-machine interfaces (HMI) for high-level process supervisory management such as issuing generator setpoint change commands. A SCADA system also allows users and operators to manage the performance of these devices from a physically remote-control center.

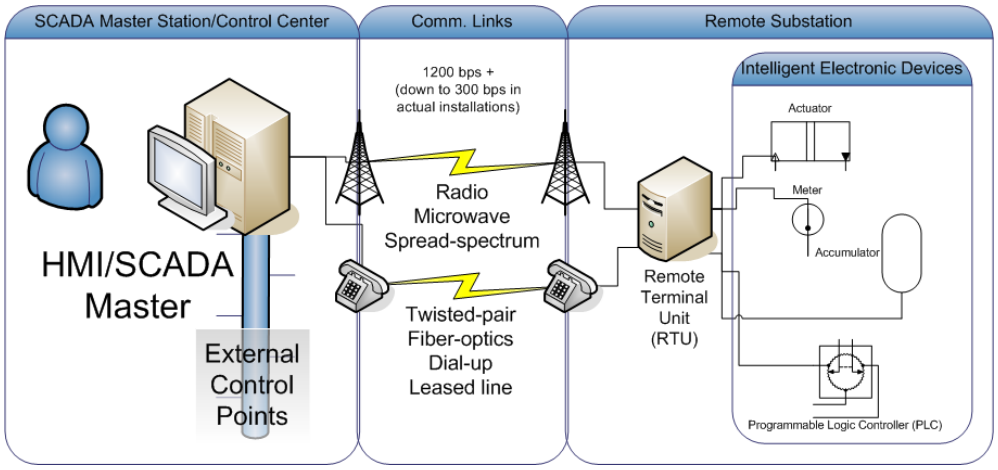

Figure 3 - SCADA Network Architecture [6]

As shown in Figure 3, a typical SCADA network consists of the following main constituent components: the supervisory system or master terminal unit (MTU), the remote terminal unit (RTU), a communications network, and field devices such as intelligent electronic devices (IEDs) [7]–[9]. The topology of the connection between the SCADA and remote terminal units varies depending on the application: it could be one MTU connected to one RTU, which is referred to as a single-master single-remote configuration [10], or several RTUs connected to one single-master MTU [11].

- ***Remote Terminal Unit (RTU)***

Remote Terminal Units are real-time microprocessor-controlled electronic devices and are the main component in SCADA systems. RTUs have direct connections (either through serial or Ethernet ports) to various sensors, meters, and actuators associated with a controlled environment. RTUs not only interface field objects to SCADA or distributed control systems (DCS) by transmitting field digital and analog telemetry data to the MTU, but also execute commands from

the master supervisory system to control field instruments and devices through actuators and switch boxes. Typically, RTU equipment has the following functionalities/controls [12]:

Table 2 - RTU Functionalities and Control Options

RTU **Functionalities/Controls**	Single indication without / with 24 / with 56 bit timestamps
	Double indication without/ with 24 / with 56 bit timestamps.
	Step position information without / with 24 / with 56 bit timestamps.
	Measured value – normalized, scaled, short floating point without / with timestamps.
	Bit strings of 32 bits without / with timestamps.
	Integrated totals (counters) without / with timestamps.
	Packed events (start & tripping) of protection equipment.
	Single commands.
	Double commands.
	Regulating step commands.
	Set point commands of various data formats.
	Bit string commands.
	Interrogation commands.
	Clock synchronization & delay acquisition commands.
	Test & reset commands.

- ***Master Terminal Unit (MTU)***

The master station or master terminal unit (MTU) is typically located in the control center, where the operator interacts with the system through a human-machine interface (HMI). The MTU polls the RTUs for data (measurements, field device statuses, etc.) by performing reading and writing operations at periodic scanning intervals. It then processes the received data and renders

various visualizations of the data. It also performs control, alarming, and networking with other nodes and sends operator-initiated command and control signals back to the field devices. A typical SCADA HMI is shown in Figure 4.

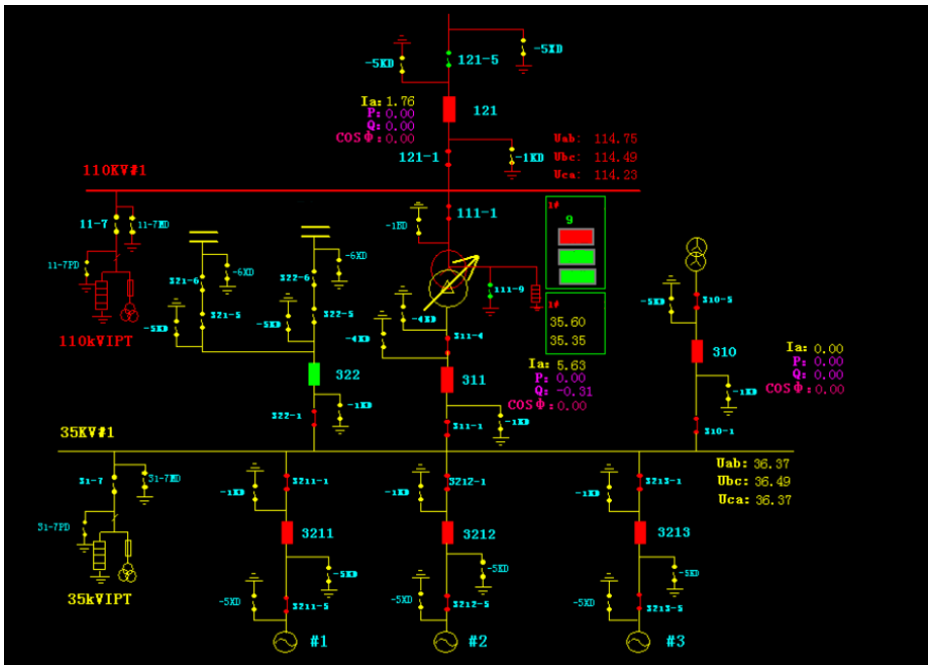

Figure 4 - A Typical SCADA HMI [13]

- ***Communications Network***

The communication network of a SCADA system transfers data among central host computer servers such as MTUs and the field data interface devices such as RTUs & control units. The medium of transfer can be cable, leased lines, Public Switched Telephone Networks, Internet Protocol (IP) based landlines, radio, microwave, satellite or any combination of these.

- ***Intelligent Electronic Devices (IEDs)***

Intelligent electronic devices (IEDs) are controllers of power system equipment. IEDs are microprocessor-based and receive data from sensors and can issue control commands, for example, to trip circuit breakers if they sense anomalies, or to raise/lower the voltage to maintain a target setpoint. IEDs include circuit breakers, protective relays, load tap changers, capacitor bank switches, etc. IEDs perform not only control functions but also protective functions. IEDs are typically designed to support some substation automation protocols such as DNP3 or IEC61850.

- ***Control Field Equipment***

Control field equipment is the actual hardware components and essentially consists of sensors and actuators. The sensors directly detect events or measure changes in a physical quantity, such as voltage or current levels, and send the information to the IEDs. Examples include current transformers (CTs), potential transformers (PTs), temperature probes, etc. [14]. Actuators, on the other hand, are responsible for moving and controlling physical devices, for example, opening, closing, activating, or deactivating a circuit breaker. Examples include breakers that energize/de-energize electrical equipment, tap-changing transformers, excitation controllers, etc. [14].

2.2.2. SCADA Architectures

SCADA systems have evolved in parallel with the growth of computing and networking technology through four generations: first generation (monolithic), second-generation (distributed), third-generation (network), fourth-generation (internet of things). It is imperative to note that SCADA systems have had to evolve as a result of growing control demands. While such evolution brings with it operational and economic advantages, the changing architecture of

SCADA systems has been a contributing factor to the growing cyber-physical security issues facing modem SCADA systems.

- ***First Generation: Monolithic***

Early SCADA systems were developed at a time when the prevalent computer technology was the mainframe computer. Hence, SCADA systems were stand-alone, hierarchical, and centralized and ran on large stand-alone mini-computers with no connectivity to other systems since networks and network services were non-existent at the time. The transmission of information between the SCADA master station and the RTUs in the field was achieved via communication protocols developed by RTU equipment vendors with the sole objective of communicating with RTUs in the field. As a result, the communication protocols used were strictly proprietary and provided only minimal functionalities to support the required scanning and controlling of points within a remote device. The communication links between the RTUs and MTUs lacked a high degree of fidelity, and thus communication security was ensured primarily through error detection and error correction codes [11]. In the event of a failure of the SCADA master station, the first-generation SCADA system used a mainframe system connected to all the RTUs as a backup system.

- ***Second Generation: Distributed***

The second generation of SCADA systems built upon advancements in system miniaturization and Local Area Networking (LAN) technology to exchange information and command processing in near real-time across multiple operating stations connected through the LAN. The single mainframe SCADA master station was replaced by multiple operating stations, each with a specific function. The operating stations shared information with each other in near real-time. Each distributed operating station served a different SCADA system function, ranging

from operating stations that served as communications processors primarily communicating with field devices such as relays and RTUs, to operating stations that provided the human-machine interface (HMI) for system operators, to operating stations that served as calculation processors.

The distribution of individual SCADA system functions across multiple operating stations provided more processing capability for the system as a whole than would have been available at a single operating station. In addition, since each operating station was a smaller and less expensive mini-computer than its predecessors, this reduced the cost compared to that of the first-generation SCADA systems. However, because the networks that connected these individual operating stations were generally based on LAN technologies and had limited reach, second-generation SCADA systems still had to be housed in a single building. Another disadvantage was that the network protocols used were still not standardized and were proprietary [11]. While this allowed a vendor to optimize its LAN protocol for real-time system applications, it limited the connection from other vendors to the SCADA LAN and thus created interoperability issues. As a result, only a few people had the technical expertise to manage and secure a SCADA installation.

- ***Third Generation: Networked***

This is the current architecture of most present-day SCADA systems. Third generation SCADA systems generally communicate using LAN and Wide Area Network (WAN) networks via the Transmission Control Protocol (TCP)/Internet Protocol (IP). Multiple networked systems that share master station functions still exist, as do vendor-proprietary protocols. However, the main difference between the second generation and third generation SCADA systems is the transition to an open system architecture [11]. As a result, SCADA infrastructures can be deployed across a WAN, which eliminates the limitation of vendor-controlled proprietary environments of the first two generations of SCADA systems. A major advantage of the networked design is that

the SCADA infrastructure can be deployed across different geographic regions and also deployed across more than one LAN network. A distributed SCADA architecture with centralized control allows for a more cost-effective and reliable large scale system. Distributing the processing across different geographical locations keeps the SCADA system operational in the event of a total loss of any one location, thus improving the entire SCADA network's reliability.

- ***Fourth Generation: Internet of Things***

Fourth-generation SCADA systems are enabled by technological advances in internet of things (IoT) technology and cloud computing. These SCADA systems compute and communicate various states in real-time by using the scalability of cloud environments to implement complex control algorithms that are practically infeasible to implement on traditional programmable logic controllers [15]. This allows for easier maintenance and integration of the fourth-generation system compared to the previous generations of SCADA systems. Additionally, the use of open network protocols such as the transport layer security (TLS) mechanisms in IoT technologies, provides a more inclusive and feasible security boundary than the varying mix of proprietary network protocols.

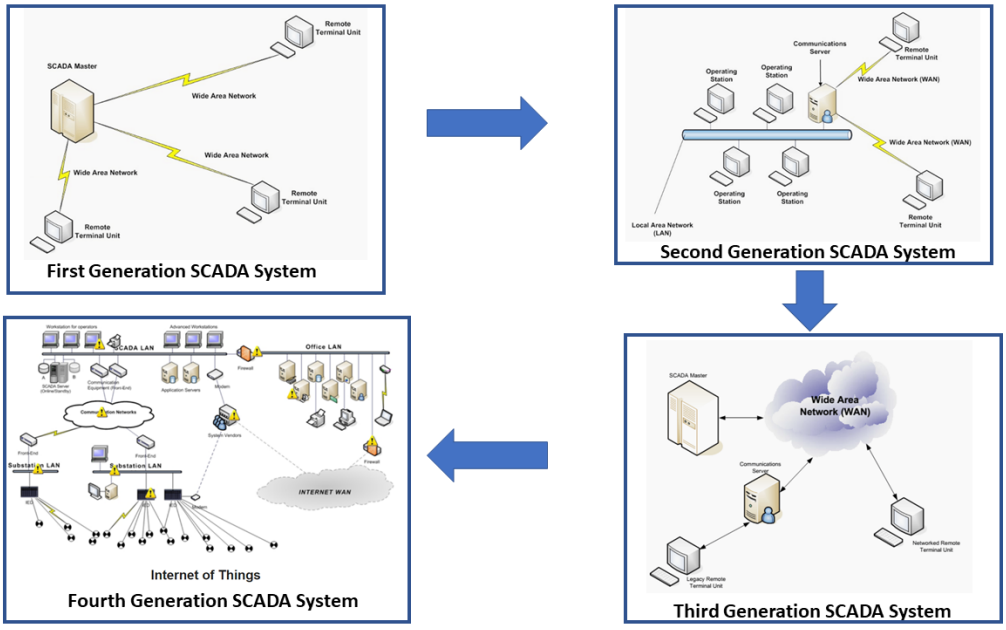

Figure 5 - Evolution of SCADA Systems [16], [17]

2.2.3. SCADA Protocols

SCADA systems need a protocol for transmitting data across multiple nodes, typically between the MTU and the RTU. According to the American Gas Association's AGA-12 standard, there are about 150-200 SCADA protocols [18]. Some proprietary vendor-specific SCADA protocols include SES-92, Modbus RTU, RP-570, and Profibus. Standard protocols that support the open standards include DNP3, IEC 61850, and IEC 60870-5-101 or 104. DNP3 is the most widely used protocol for data transport in electricity systems in the US. The work done in this dissertation uses the DNP3 protocol in the implementation of a cyber-physical security assessment tool. Many of the standardized protocols are designed to operate over TCP/IP. However, it is a good security measure to avoid connecting SCADA systems to the internet to reduce the cyber threat surface area. DNP is primarily used in North and South America, parts of Asia, South Africa and Australia, while IEC 60870-5-101 or T101 is strongly supported in Europe [12].

Table 3- Some Common SCADA protocols [11]

Protocol	Organization	Common Industries	Features
DNP3	Developed by GE Harris, Managed by the DNP organization	Electric Utilities, Gas Distribution, and Water distribution	Object-Oriented. Three-layer OSI model. Open non-proprietary standard
Modbus (Modbus/TCP)	Developed by Modicon	Gas and Oil and electric substations, transportation	Initially developed for modicon's PLCs. Open standard and royalty-free. Simple to implement. Both serial and TCO version are available. Simplicity and wide use make this an excellent protocol, when integrating multiple application
Ethernet/IP (Industrial Protocol)	Open DeviceNet Vendors Association (ODVA)	Industrial Automation	
DeviceNet	Open DeviceNet Vendors Association (ODVA)	Industrial Automation	Uses CAN as its backbone and, originally developed by allen-bradley. Supports master-slave as well as peer to peer
IEC 60870-5	IEC TC57		
IEC 61850	IEC TC57	Substation automation, distribution automation	Ultra-fast response times

2.3. Control and Monitoring of Energy Management System

The supervisory control and data acquisition (SCADA) system monitors the state of the electric power through data transmitted by remote control units (RTUs) to the control center every

2-5 seconds. The received data is processed by the state estimator, and the output provides energy management system (EMS) operators with situational awareness of the entire system's current operating conditions despite noisy or corrupted measurements. State estimation is one of the vital components in the Energy Management Systems (EMS) of power system control centers [19],[20]. State estimation involves the optimal estimation of the power system state by using data from power meters, sensors, smart meters and system parameters. State estimation also includes algorithms for bad data detection and network topology checking to offer robustness against sensor faults and line contingencies. State estimation is at the foundation of system monitoring, protection, and control. The output of the state estimator is employed for different applications of EMS - optimal economic dispatch, contingency analysis, and system operational security assessment.

The electric grid cannot reliably operate without a valid state estimator solution for extended periods of time. The 2003 Northeast blackout in the U.S was caused in part by convergence issues in the state estimator software as a result of topology errors [21]. The errors caused the estimator's solution to diverge and thus rendered all downstream analysis tools like Real-Time Contingency Analysis unusable for hours. This incident emphasizes the importance of having an accurate network topology in the EMS. Equally important is having redundancy. Redundancy in measurements is important not only for obtaining a valid state estimation (SE) solution, but also for identifying bad data and topological errors. Although utilities and ISOs maintain high redundancy in measurements, there could be cases such as system islanding or issues in data acquisition systems that result from faulty sensors or downed communication links and can reduce redundancy to critical levels. In such cases, total system observability cannot be guaranteed [22]. The problem of the detection and correction of topology errors has been studied for a long

time, but the high exposure of the current SCADA infrastructure and the EMS to cyber threats has added a new dimension to the problem.

2.4. Bulk Electric System Cyber-Physical Security Threats and Vulnerabilities

This section discusses the current cyber-physical security threats and vulnerabilities faced today by Supervisory Control and Data Acquisition (SCADA) systems as well as the Energy Management Systems that contain the computational engine of grid essential algorithms and functions essential to a bulk electric delivery system. Several vulnerabilities in SCADA and EMS systems can be exploited by various attack vectors, as shown in Figure 6.

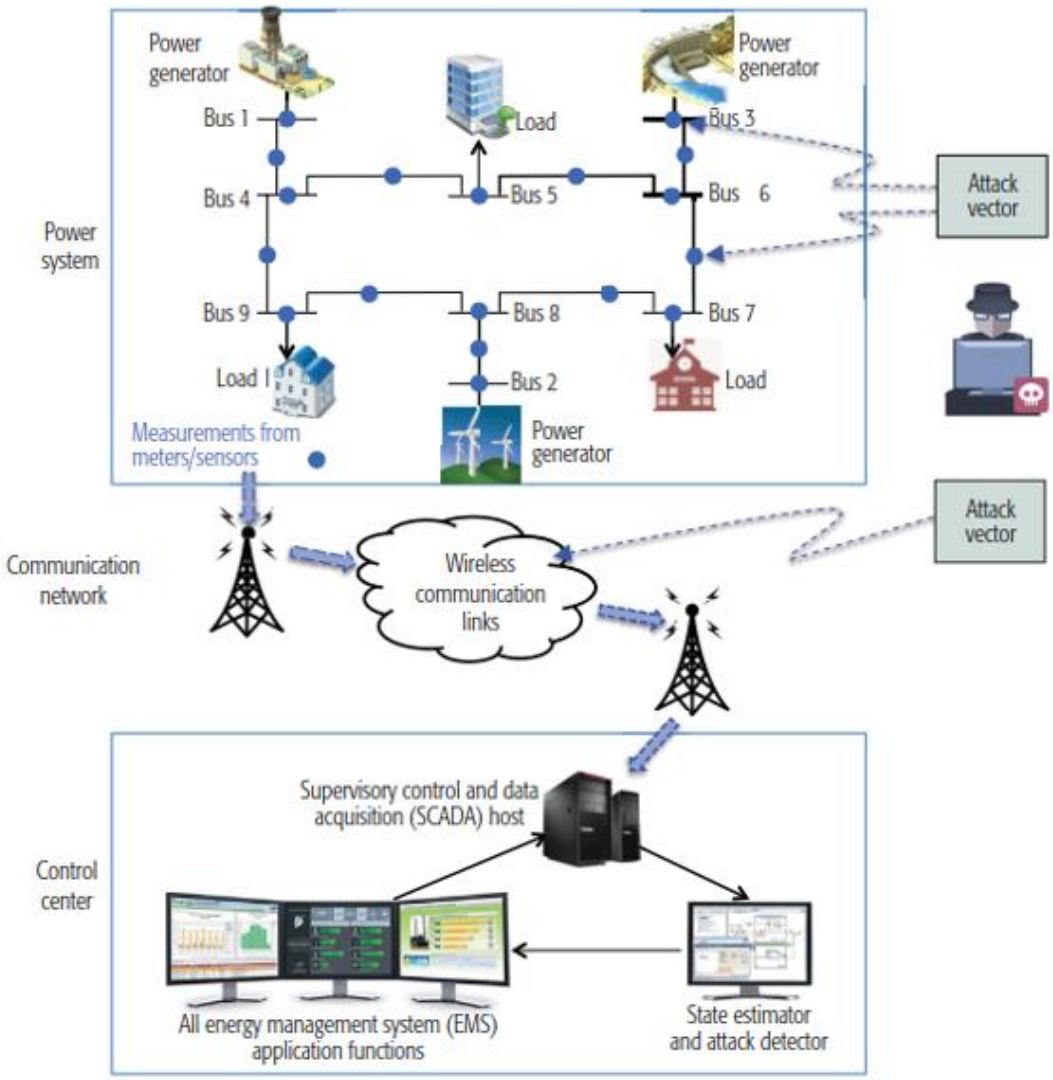

Figure 6 - Vulnerabilities in the Power Grid Exploited by Attack Vectors [23]

Table 4 - Threats against SCADA Systems [11]

Threat	Description
Viruses	A virus is a type of malicious software that replicates itself by modifying other non-malicious computer programs and inserting its own code when executed. The 2015 and 2016 attack by the Stuxnet virus shows that an attack on a SCADA system is particularly attractive for hackers because of the significant physical damage the attack can cause.
Worms	Computer worms are stand-alone malware programs that replicate themselves through networked computers by exploiting software or hardware vulnerability. Worms that target electrical systems could disrupt electricity delivery operations.

Bot-network operators	Bot-network operators are a network of compromised computers infected with malicious software and controlled as a group without the owner's knowledge. Botnets can be used to carry out DDOS attacks to overload a SCADA network with requests, thus stretching out network resources and rendering the network inaccessible to its intended users.
Insider threats (insiders)	An insider threat is a security risk that comes from within the targeted system or organization. Insider threat can include employees, former employees, contractors, or business associates who have inside information about the organization's practices and protocols. Insiders have been the main source of computer crime
Hackers	Hackers use computers to gain unauthorized access to data. Hackers can exploit SCADA vulnerabilities to take full control of critical infrastructure.
Criminal Groups	Criminal groups such as hostile nation-states can launch cyber warfare to create devastating attacks on SCADA systems in the US by exploiting known and unknown vulnerabilities in these systems. This can be done through several means, some of which include phishing emails and network reconnaissance by analyzing publicly available information.

2.4.1. SCADA System Vulnerabilities

In recent years, cyber-attacks on SCADA and EMS systems such as the Maroochy Shire sewage spill attack in January 2000 [24], the Davis-Beese Ohio nuclear power plant attack, the Slammer Worm attack in January 2003 [24], the Stuxnet Worm attack on Iranian Nuclear Facilities in November 2010 [25], the cyber-attacks on Ukraine's power grid in December 2015 and February 2016 [26], which left 80,000 people in the dark for six hours, have brought more attention to BES vulnerabilities, threats, and impacts of cyber-attack on critical power delivery systems.

The biggest cyber threat to BES and SCADA systems is unauthorized access by an insider or disgruntled ex-employee leading to a cyber incident. The United States Computer Emergency Readiness Team (US-CERT) released a vulnerability advisory [27] warning that unauthenticated users could download sensitive configuration information including password hashes from an

Inductive Automation Ignition system by utilizing a standard attack type that leverages access to the Tomcat Embedded Web server. Additional threats to unauthorized access to BES and SCADA systems are publicly released worms/viruses/Trojans, as described in Table 4, constitute major threats to BES and SCADA systems. The information presented in Table 4 was adapted from the Department of Homeland Security ICS-CERT's cyber threat source descriptions presented in [28]. In June 2010, the anti-virus security company VirusBlokAda reported the first detection of malware that attacks SCADA systems running on the Windows operating systems. The malware is called Stuxnet and uses four zero-day attacks to install a rootkit, which in turn, logs into the SCADA's database and steals design and control files [29], [30]. The malware is also capable of changing the control system and hiding those changes [31]. A vulnerability is a weakness of a system that can be used by an adversary, such as an attacker, to compromise one or more of its attributes by crossing priviledge boundaries. The most common threat vectors to BES and SCADA systems are described in Table 4. Threats and vulnerabilities in SCADA systems are both historic and new as a result of the incorporation of new technology and equipment, as shown in Table 5.

Table 5 - Table of SCADA Equipment Vulnerabilities [32]

Equipment	Vulnerable Point of Access	Risk
Protective devices w/o remote access (e.g. relays, IEDs, PLCs, reclosers)	• Local access to protective devices • Local access to protection settings	• Protective equipment accidentally or deliberately damaged • Protection settings accidentally or deliberately altered
Protective devices with remote phone access	• Electronic access to protective devices via modem or codec • Electronic access to protection settings	• Dial-in number accessible via social engineering or automated modem scan • Access control circumvented by password attack • Protection settings accidentally or deliberately altered
Protective devices with remote network access	• Electronic access to protective devices via system port or network address • Electronic access to protection settings • Electronic access to data packets • Equipment vulnerable to Denial of Service (DOS) attacks	• Network address accessible via social engineering or automated network port/IP scan • Access control circumvented by password attack • Protection settings accidentally or deliberately altered • Data packets visible on the network • Equipment inaccessible, and possibly non-functional, during DOS attacks
SCADA equipment with remote access via private network	• Physical access to SCADA system • Electronic access to subordinate protection equipment • Electronic access to protection settings	• SCADA system accidentally or deliberately damaged • SCADA functions accidentally or deliberately altered • Protection settings accidentally or deliberately altered
SCADA equipment with remote phone access	• Electronic access to SCADA system via modem or codec • Electronic access to subordinate protection equipment • Electronic access to protection settings	• Dial-in number accessible via social engineering or automated modem scan • Access control circumvented by password attack • SCADA functions accidentally or deliberately altered • Protection settings accidentally or deliberately altered
SCADA equipment with remote network access	• Electronic access to SCADA system via system port or network address • Electronic access to control and data packets • Electronic access to subordinate protection equipment • Electronic access to protection settings • SCADA vulnerable to DOS attacks	• Network address accessible via social engineering or automated network port/IP scan • Access control circumvented by password attack • SCADA functions accidentally or deliberately altered • Protection settings accidentally or deliberately altered • Control and data packets visible through a network sniffer • SCADA inaccessible, and possibly non-functional, during DOS attacks

Traditional IT security breaches have targeted data and personal information. However, ICS cyber attacks target physical processes and assets, specifically the control systems and

intelligent electronic devices (IEDs) connected to the electric grid [33]. Since the 1980s, there have been almost 800 total ICS cyber incidents globally documented. More than 250 of these cyber attacks occurred in 2013, with over 50% of them occurring in electric utilities in North America. It is believed that the reported numbers understate the actual cyber attack attempts and cyber incidents, primarily because of inconsistent and voluntary ICS cyber incident tracking and reporting [34]. As a result, the security of some SCADA-based systems has come into question as they are seen as vulnerable to cyber-attacks [24], [35], [36]. More specifically, security researchers are concerned about the following issues:

- The lack of concern about security and authentication in the design, implementation, and operation of some existing SCADA infrastructures
- The belief that SCADA systems have the benefit of security through obscurity through the use of specialized protocols and proprietary interfaces
- The assumption that SCADA infrastructures are secure because they are physically secured
- The assumption that SCADA networks are secure because they are disconnected from the internet

2.4.2. Energy Management System Threats and Vulnerabilities

Vulnerabilities in today's electric grid expose the grid to cyber-physical security issues. A contributing factor is the ongoing transition of electricity control infrastructure to a cyber-controlled paradigm. Figure 9 illustrates the traditional real-time control loop implemented through EMS systems for bulk electricity delivery. Complex decisions are made automatically through the automatic generation control (AGC) system and through human-in-loop (HIL) operator decision making. Figure 9 represents the operational paradigm that has successfully addressed electricity system operations for several decades. The EMS system ultimately

implements a sophisticated control loop of sensing, estimation, optimization and actuation, which can be perturbed at any stage by cyber-physical attacks. The EMS can provide different levels of feedback, from SCADA-only operation to instantaneous, deterministic optimization through the security-constrained optimal power flow (SCOPF). Except for instances of AGC control, all the control centers require HIL operation at the core of operational decisions.

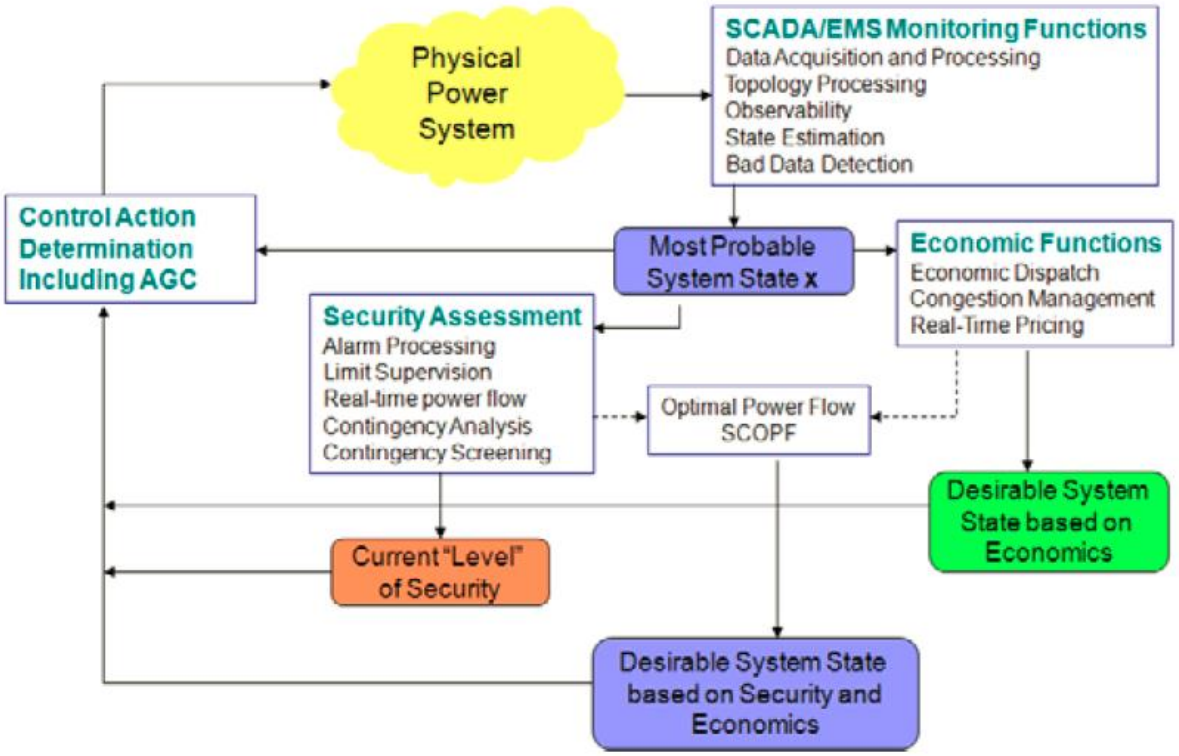

Figure 7 - Control loop operation of Energy Management System

The authors in [37] published one of the earlier works on cyber-physical attacks, specifically bad data injection on power systems' state estimation. Since then, research interest in this area has increased significantly, and several research efforts addressing challenging research problems in this domain have been published. The authors in [6] proposed a unified formulation for the problem

of constructing attack vectors for linearized measurement models and, using this formulation, designed a new low-complexity attacking strategy that significantly outperforms naive relaxation. The results demonstrated that it is possible to defend against malicious data injection if a small subset of measurements can be made hardened or insusceptible to the attacks. However, selecting such subsets is a high-complexity combinatorial problem given the typically large size of electrical grids. In [38], the authors explored the effects of data integrity attacks on voltage control devices like FACTS, SVC, etc. In [39], a graph theoretic efficient algorithm with polynomial-time complexity is designed to implement an unobservable malicious data attack. When the unobservable attack can not be implemented due to limitations of meter access, attacks are constructed to minimize the residue energy of the attack while guaranteeing a certain level of increase of mean square error. Research works exploring the vulnerabilities and threats to EMS have been published, with a majority focusing on the simulation of different types of attacks on the DC state estimator. Other works have explored cyber-attacks on power system operations like AGC, voltage control, state estimation, distribution applications like AMI, and infrastructure elements like PDCs.

The authors of [40] implemented a game-theoretic approach to launch attacks in which system operators in charge of defending a system act as malicious actors willing to attack the system. Work in [22] explores designing and implementing an unobservable attack on a DC state estimator. In [41], the authors studied the effect of bad data injection through linear state estimators under the assumption that the attacker has perfect knowledge of the power systems topology. Work done in [42] determined how many critical measurements attackers needed to have access to to minimize their efforts in carrying out an undetectable bad data injection attack. In [43], the authors characterized various attack models that took into account both the attacker's perfect and imperfect

knowledge of the system configuration. In [44], the authors studied economic dispatch infeasibility as a result of a cyber-attack by exploring two scenarios to bad data injection and its effect on economic dispatch: an opportunistic attack scenario in which an adversary launches an attack in an appropriate instance, and a dynamic attack, in which economic dispatch drifts to an undesirable state gradually. The authors in [45] showed how a bad data injection attack affects optimal power flow (OPF) and could bias the system resulting in higher operations costs. In [38], the authors study data injection attacks on state estimation under two scenarios: random data injection attacks and targeted data injection attacks on specific measurements. The results of this work presented the probability of finding a suitable attack vector and the computational time required for finding an attack vector.

Bad data injection attack via an SQL-injection on PMU Data Concentrators is considered in [39], [46]. Work in [47] explored the impact of data integrity attacks on the AGC system operation and quantify the attack impacts in terms of load-generation imbalance and frequency violations. The authors in [48] present a false data injection attack against the state estimation in deregulated electricity markets. The results showed that a class of attacks could bypass the bad data measurement detecting and lead to profitable financial misconduct by affecting the market clearing prices (LMP). Several recent works have explored various aspects of cyber-attacks on EMS system function that impact reliable EMS operation. These include modeling attacks, minimizing the number of compromised meters for PMUs, and various detection and mitigation strategies [49], [50], [39], [51], [52], [52], [53].

2.5. Securing Bulk Electric Systems

Prior efforts and initiatives by several entities to secure BES systems focused on reliability, i.e., protecting the system against faults. These entities included balance authorities, utilities, independent system operators (ISOs), regional transmission organizations (RTOs), the Federal Energy Regulatory Commission (FERC), and the North American Electric Reliability Corporation (NERC), etc. However, in recent years, these initiatives have expanded to consider and improve the system's cyber-physical security. The section discusses three types of initiatives designed specifically for securing SCADA and EMS systems: technical, industrial, and government-led.

Technical Approaches

Several mechanisms have been developed to secure BES transmission assets, such as EMS and SCADA systems. One such mechanism deploys phasor measurement units (PMUs) at different locations to protect the system from attackers in advance. Because PMUs measure voltages and currents on a power grid by using a common time source that is based on global positioning system (GPS) time, they can provide accurately time-stamped measurements for geographically dispersed nodes. Another mechanism develops and implements advanced signal processing techniques at the control center to identify bad data injection attacks [54]–[56].

Industrial Approaches

- **The North American Electric Reliability Corporation (NERC)**:

NERC is a nonprofit self-regulatory authority that develops and enforces standards to ensure the reliability and security of the bulk electric system in North America [57]. The Federal Energy Regulatory Commission (FERC), on the other hand, is the agency that regulates both the transmission and wholesale sale of electricity and natural gas in interstate commerce as well as the operations of regional markets. As a result of growing threats to the energy sector from cyberspace,

the security of BES is no longer a certainty. In response to these threats, NERC has developed the Critical Infrastructure Protection (CIP) Reliability Standards and submitted them to the FERC for approval. The CIP Reliability Standards require certain users, owners, and operators of a BES to comply with specific requirements to safeguard critical cyber assets [58]. NERC Standards CIP-002 through CIP-011, as summarized in Figure 8, provide a cybersecurity framework for identifying and protecting Critical Cyber Assets to support reliable operation of the Bulk Electric System (BES).

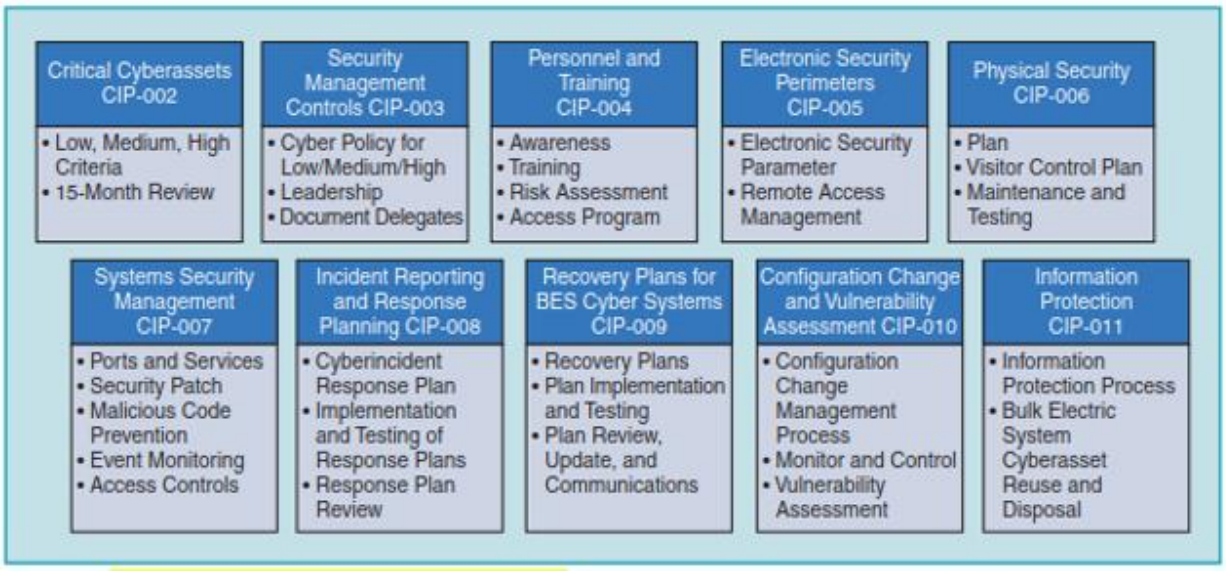

Figure 8 - Current NERC CIP Standards as of March 2016 [34]

These standards recognize the differing roles of each entity in the operation of the BES, the criticality and vulnerability of the assets needed to manage BES reliability, and the risks to which they are exposed [59].

- **The American Gas Association (AGA)**:

  AGA has developed practices, protocols, and processes required to protect SCADA communications against cyber incidents [60]. The main focus of the AGA security initiatives is to ensure the confidentiality of SCADA communications.

Government-Led Approaches

- **The Department of Energy (DOE):**

  The DOE has established the national SCADA testbed program [INL] and developed a 10-year roadmap for securing control systems in the energy sector [61] with the following four main objectives: (1) measure current security, (2) develop and integrate protective measures, (3) detect intrusion and implement response strategies, and (4) sustain security improvements.

- **Sandia National Laboratories:**

  Sandia established the Center for Control System Security with the objective of modeling, designing, simulating, verifying, and validating problems and challenges encountered in real-world critical infrastructures [62]. The center's findings are integrated into research efforts that focus on solving current control system security problems and developing next-generation control systems.

- **Working Groups**:

  Working groups, such as the collaboration between Hart and the ISA, have been established to facilitate the adoption, implementation, and standardization of wireless sensor networks in SCADA control systems. They also work to configure hop-by-hop and end-to-end confidentiality and integrity mechanism in the wireless communication protocol while providing the necessary protocols for access control and key management [61], [63].

A combination of authentication, encryption, and access control have been employed as the main techniques in electric power systems to safeguard IEDs, PLCs, RTUs, controllers, communication processors, and SCADA systems against cyber-attacks. Several cyber-attack mitigation tools implementing a combination of these techniques. Table 6 captures a snapshot of technologies to safeguard SCADA network equipment [32].

Table 6 - Mitigation Tools and Techniques [32]

Tool/Process	**Purpose**	**Cost**	**Scalable?–Programmable?**	**Ease of Use**
Device-based passwords/Pins	Access control	NA	Via custom programming.	Use and programming trivial.
Password generators	Software that generates strong passwords	Free–low cost	Via custom programming	Trivial to use; nontrivial to program.
Audit logs	Record device or system access	NA	Via system features or custom programming	Trivial to use; nontrivial to program.
ID devices	Hardware authentication	$2–$100 per unit; > $1000 for network servers	Via server license & host platform.	Trivial to use; nontrivial to program.
Biometrics	Strong single-factor ID, or two-factor ID	$200–$1000	Via custom programming on host.	Trivial to use; nontrivial to program.
Modem key/lock	Secure modem connections	$150 per pair	Any number of keys per lock; key codes are user selectable.	Use and programming trivial.
Secure modems	A. Programmable Secure Modem B. Encrypting Modems	$250–$600 each	A. Programmable user accounts that validate incoming calls. B. Handshaking security that works only in pairs.	Trivial to use; programming difficulty varies.
Virtual Private Network (VPN) devices	Hardware network security	~$2000 pair	Only work in pairs	Use and programming nontrivial
Firewalls	A. Software network filter B. Hardware network gates	A. Free–$5000+ B. $1200–$26000	NA	Use and programming nontrivial
Intrusion Detection Systems (IDS)	System anomaly and/or intrusion signature detection	Free– $50000	Few are scalable	Use and programming nontrivial
Public Key Infrastructure (PKI)	Authentication and secure network communications	Free– $10000	Inherently scalable	Use and programming nontrivial

Once the vulnerabilities of modem SCADA systems for bulk electric power systems have been explored, they then need to be secured/protected. One approach that has been applied to protecting

SCADA systems is to deploy traditional network security technologies. However, because SCADA systems and traditional IT systems are different, applying traditional IT cyber-security measures to SCADA systems is often ineffective or simply inapplicable. The differences between traditional IT systems and SCADA/EMS systems were first presented in [11] and are re-adapted here in Table 7, as shown below.

Table 7 - Difference between Traditional IT Systems and SCADA

Category	**Traditional Information Technology Systems**	**SCADA Systems**
Performance	High throughput can tolerate delay and jitter	Medium to low throughput cannot tolerate delay or jitter
Focus of Security Architecture	Protection is focused on the central core of the system, so-called "hard in the middle and soft on the outside"	Need to protect the edges or perimeter devices such as RTUs and field devices. Also need to protect core internal systems as well
Priority of security primitives (Priority in ascending order)	Confidentiality Integrity Availability	Availability Integrity Confidentiality
Component Lifetime	3 – 5 years	15 – 20 years
Physical Accessibility	Easily accessible	Isolated and remote, may be very difficult to access
Protection focus	Protect data – intellectual property, credit card, personal identifiable information (PII)	Protect process – protect the protection, control, and monitoring process
Communication protocols	TCP/IP Suite – TCP, IP, UDP, DNS, DHCP etc. ISO27000	Over 1000 protocols – Most popular are Modbus, DNP3, PROFINET.PROFIBUS, OPC etc.
Control mechanism	Programmable Logic Controllers not common	Programmer Logic Controllers are the backbone of SCADA/ICS Systems control process [64]
System downtime	Tolerated	Not acceptable
Interoperability	Not Critical	Critical
Computing resources	"Unlimited"	Very limited

2.5.1. Securing SCADA and EMS with standard IT Technologies

Two ways to keep SCADA systems secure are to provide good network segmentation by firewalls or with a virtual LAN (VLAN) [11]. In [38], Munshi discusses applying encryption to protect against spoofing commands and access to data of field equipment such as PLCs and RTUs; employing IPSEC to protect unsecured communications that may be traveling over unsecured shared networks; adopting operating system hardening, patch management, network equipment access control, server access controls, physical security, virus protection strategy, and user authorization to prevent access to the control center; using network controls like firewalls, proxy servers, and network segmentation to protect against unscrupulous access to remote SCADA clients, ERP systems, and corporate users of SCADA data, and web services.

The use of modems for remote access to SCADA systems makes it an easy target by creating multiple network entry points. Password-protected modems and encrypting modems were solutions recommended by Oman, et al. [32]. Permann and Rohde [11] discussed using well-established traditional IT network scanning and vulnerability assessment tools like Nmap, Nessus, and Ethereal for security assessment of SCADA control systems. Dolezilek et al. explored secure substation information system installation that uses standard IT technologies in [65]. Abshier [66] summarized ten important design and process principles for securing control systems: governance, security awareness and training, policies and procedures, change management, security architecture, adding devices and remote access, vulnerability, risk assessment and penetration tools, incident response, configuration and patch management, and monitoring. Some additional strategies for building a security plan for securing SCADA systems with traditional IT technologies are given in [67]. Comprehensive guidelines for designing and implementing secure

SCADA systems and control networks are also being developed by several industry organizations and commissions.

2.6. Current SCADA and EMS Cyber-Physical Research Challenges

Over the past several years, industry groups and academics have begun to work towards addressing SCADA security issues. This can be seen in the increasing number of publications related to SCADA security [68], [69]. The following are a number of identified research challenges in the area of cyber-physical security for bulk electric systems:

- Improve access controls to SCADA networks to make it harder for attackers to gain access to the SCADA network.
- Improve security inside SCADA networks, including developing efficient monitoring tools that make carrying out an attack difficult.
- Improve the security management of the SCADA network.
- Develop advanced signal processing techniques at the control center to identify bad data injection attacks
- Develop advanced statistical methods for efficient and reliable bad data detection
- Design correction techniques for bad data injection attacks

Solutions to these challenges must take into consideration the unique demands of SCADA and EMS systems, and these challenges will be resolved through a combination of developing better trust management protocols, advanced algorithms for bad data detection, and novel co-simulation paradigms and platforms that allow us to better understand system dynamics during a cyber-physical attack

2.7. The Need for Bulk Electric System Cyber-Physical Security Co-Simulation

Efforts by research and industry to design cyber-physically secure power grids are constrained by the availability of realistic cyber-physical environments. Hence, the research and development of cyber-physically secure grids will depend heavily on the availability of representative environments such as co-simulators and testbeds where current solutions and future ideas can be implemented, tested, and extensively verified. Real-world applications require prototype implementations. This allows for safe and fast verification of concepts that can facilitate research results that can be transferred to the power system industry. Co-simulators and testbeds which integrate SCADA hardware, EMS software, and emulation and simulation techniques to provide an accurate electric grid cyber-physical infrastructure need to be developed in the following areas:

1. Cyber-Physical Security Attack Models Development
2. Cyber-Physical Attack Impact Analysis
3. Cyber-Physical Security Metrics
4. Enhanced Operator Visualization
5. Mitigation Research
6. Vulnerability Research
7. Security Validation
8. Interoperability
9. Cyber Forensics
10. Operator Training

The work in this dissertation focuses on objectives (1) – (4). One of the earliest research efforts to design a security-oriented testbed that addressed some of the objectives above is presented in [70]. Sandia National Laboratory developed the ***Virtual Control System Environment (VCSE)*** to

investigate SCADA vulnerabilities of energy systems [70]. The goal of the testbed was to enhance wide-area situation awareness by developing and analyzing possible cyber attacks. The power system simulator was integrated with simulated RTUs and HMIs to emulate cyber-attack scenarios such as MIMTM, insider attacks. VCSE utilized the DNP3 and Modbus protocols and uses OPNET System-in-the-Loop emulation to allow for the integration of network devices into the simulated network. The tool was designed to provide support for operator training, vulnerability exploration, mitigation development, and attack evaluation. Table 8 below provides a list of security-oriented simulators and testbeds that have been built to date.

Today, we have access to a number of power system simulators: PowerFactory, PowerWorld, DigSilent, Matlab/Simulink. Communication network Simulators include OPNET, OMNET++, NS2, NS3 as well as hybrid simulators such as GridSim. Most platforms involve the combination of several dedicated simulators or hybrid platforms to achieve a co-simulation environment. In some of the testbeds, actual data acquisition and sensor/actuator components are integrated with power system simulators by using middleware to enable hardware-in-the-loop (HIL) simulations.

Table 8 - Existing Cyber-Physical Security Testbeds, Simulators and Co-Simulator

Testbed/ Simulator Name	**Year Published**	**Targeted Research Area**	**Covered Electric Grid Domains**	**Test Platform**	**Communication Protocols**	**Technology**	**Network Type**	**Reference**
Sandia Lab (VCSE)	2008	Wide Area Situational Awareness Cyber Security	Transmission Operations	Simulator	DNP3.0, Modbus	Ethernet, RS232	WAN, LAN	[70]

Table 8 continued

Virtual Power System Testbed	2009	Cyber Security	Transmission Operations	Simulator	DNP3.0	Ethernet	WAN, LAN	[71]
Florida International University Smart Gris Testbed	2011	Wide Area Situational Awareness Distribution Grid Management Network Communications Cyber Security	Transmission Operations Customer	Hardware	IEC 619850, C37.118, Modbus, DNP3.0, OPC UA	Ethernet, RS232, Cloud	WAN, LAN	[72], [73]
TASSCS	2011	Cyber Security	Transmission Operations	Simulator	DNP3.0, Modbus	Ethernet	WAN, LAN	[74]
University College Dublin	2011	Cyber Security	Transmission Operations	Hybrid	-	-	-	[75]
SCADASim	2011	Cyber Security	Operations Service Provider	Simulator	DNP3.0, Modbus	Ethernet	WAN, LAN	[76]
SCADA Security Lab	2011	Cyber Security Network Communications	Transmission Operations	Real-Time Simulator	IEC 619850, C37.118, Modbus, DNP3.0	Ethernet, RS232, Cloud	WAN, LAN	[77]
DeterLab	2012	Cyber Security	-	Hardware	N/A	Ethernet	WAN, LAN	[78]
PowerCyber Testbed	2013	Wide Area Situational Awareness Cyber Security	Transmission Operations	Real-Time Simulator	IEC 619850, C37.118, Modbus, DNP3.0, OPC UA	Ethernet	WAN, LAN	[79]
Texas A&M	2014	Wide Area Situational Awareness Cyber Security	Transmission Operations	Real-Time Simulator	-	-	-	[80], [81]

Table 8 continued

Cyber-Physical Testbed	2015	Wide Area Situational Awareness Cyber Security	Transmission Operations	Real-Time Simulator	IEC 619850, C37.118, DNP3.0	Ethernet	WAN	[82], [83]
VSCADA	2015	Wide-Area Situational Awareness Cyber Security	Transmission Operations	Simulator	OPC DA	Ethernet	WAN	[84]
Network Intrusion Detection System	2015	Cyber Security Network Communications	Operations	Hybrid	Modbus	Ethernet	WAN, LAN	[85]
State University of New York	2015	Wide Area Situational Awareness Cyber Security	Transmission Operations	Hybrid	C37.118, OPC DA	Ethernet, Wi-Fi, Cloud	WAN	[86]
Cybersecurity Testbed for IEC61850	2015	Cyber Security Network Communications	Operations	Real-Time Simulator	IEC 61850, C37.118	Ethernet	LAN	[87]
INEL	2015	Wide Area Situational Awareness Cyber Security	Operations	Hardware	-	-	-	[88]
Georgia Tech. ACES CPSA	2017	Cyber-Physical Security Wide Area Situational Awareness	Transmission Operations	Co-Simulator	DNP3.0, C37.118	Ethernet	WAN	[89]

CHAPTER III

GRID CYBER-PHYSICAL SECURITY MODEL

3.1. Electric Grid Cyber-Physical Security

Electric grid cyber-physical security can be defined as the ability of the grid to not be operationally disrupted or physically damaged through a cyber-attack. We adopt a cyber-physical system (CPS) and Internet of Things (IoT) devices approach, one that tightly couples computing, information, and communication technologies (ICT) with physical power system apparatuses and system operations. Cyber-physical security presents new concerns that require new approaches and solutions. Traditional cyber-security approaches and measures are either deficient, insufficiently scalable, unsuitable, or simply inadequate to tackle the challenges posed by the highly complex cyber-physical security concerns in emerging grid environments [90].

Cyber-physical attacks attempt to perturb grid operations by compromising the cyber layer by incapacitating communications devices and/or making communications resources unavailable [91]. This can cause disruptions in the topology of the communication network, communication and controlling devices in the network and in the field, which impacts the following aspects of communication performance:

a) Link baud rate.

b) Propagation time or delay.

c) Maximum number of packets that can be sent without major collision or packet dropping.

d) Maximum allowable size of each packet.

However, the effect of these attacks transcends the cyber layer, as the goal of cyber-physical attacks is to actually compromise power system operations and damage power devices. Grid cyber-physical attack mechanisms range from traditional cyber-attacks, such as man-in-the-middle [92],

denial-of-service [93], replay [94], and impersonation [95], to bad data injection, malicious command injection, and coordinated denial-of-service on Remote Terminal Units (RTUs). The current state and overall health of the power system can also be affected by attacks over the communication network, including delay attacks, synchronous flood attacks, and distributed denial-of-service attacks on devices. During these attacks, the power system may undergo various state transitions and eventually enter an operationally insecure state, in which various physical quantities such as power flows exceed established operating limits, voltages are lower than normal, and some loads are disconnected.

It is physically impossible to protect all the assets of a large complex cyber-physical system such as the electric grid against a cyber-physical security attack. Consequently, there arises the need to develop methods, algorithms, and measures that can detect and counter cyber-physical attacks. Traditional IT cyber-security, although mature, is not equipped to capture the potential consequences of attacks on physical systems. Although system theory deals with concepts such as the performance, stability, and safety of physical systems, its theoretical framework, while well consolidated, does not provide a model applicable to cyber-physical systems. As a result, a new security concept, namely cyber-physical security is required. By bringing together cybersecurity and system theory, Cyber-Physical Security makes possible, a holistic security assessment of a cyber-physical system. Unlike the traditional security assessment of just a single domain, i.e., either the power system operational security alone or the communication network cyber-security alone, Cyber-Physical Security is capable of integrating dynamic systems and threat models within a unified framework.

3.2. Cyber-Physical Security Attack Modeling

We develop a framework for cyber-physical security attack modeling, as shown in Figure 9. The two main approaches to cyber-physical attacks on bulk electric systems are bad command injection and bad data injection. The model developed in this dissertation is a feedback control model in which a bad command injected into the BES is executed by actuators, which can have serious consequences for the operation of the system. The second aspect explores the attacker manipulating legitimate sensor measurements through a bad data injection attack. The details of the bad data injection attack are discussed in the next chapter, while this chapter explores the functional requirements and use case scenarios for bad command injection simulation, which will be further discussed in later chapters.

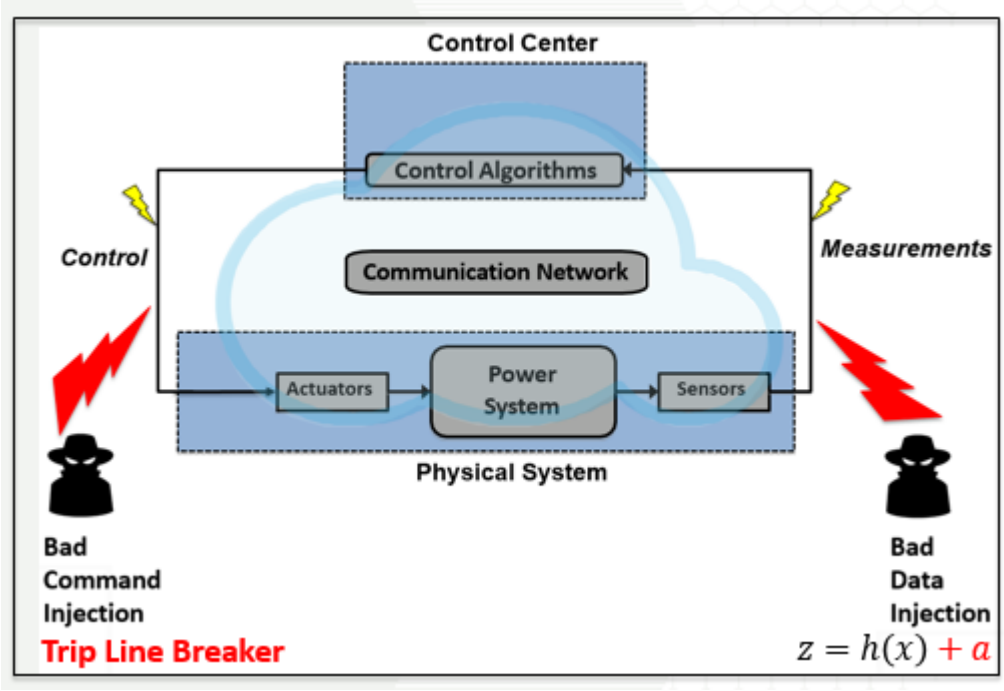

Figure 9 - Cyber-Physical Security Attack Model

3.3. Functional Requirements of Cyber-Physical Security Assessment (CPSA) Co-simulator

Technically, the tool must contribute to research and educational learning as well as technical and non-technical analysis by electric utility staff. The tool should do the following:

1. Provide monitoring and situational awareness capabilities to the operators and system administrators.
2. Detect and analyze possible electricity disruptions that can occur in the system as a result of a specific cyber-attack.
3. Enhance the security and resilience of the power system by identifying and assessing the state of the cyber-physical system and then suggest appropriate operator actions under attack scenarios.
4. Generate historical logs and cyber-physical security metrics for different power and communication components and identify weak elements in the system to help operators respond quickly when a similar situation occurs at other locations.
5. Apply user-generated rules for the normal operating range to provide a better understanding of the behavior and normal state of the integrated cyber-physical system.

3.4. Use Case Scenarios for CPSA

A malicious actor can perform a malicious command injection attack by sending a false control command to a substation RTU. If the adversary does not have complete knowledge of the system and simply injects a false command at random, the operator should be able to identify and stop the execution of the malicious command on the power system. However, if the adversary has partial or complete knowledge of the system, it can purposefully inject a specific malicious command to maximize the damage in the system at large. One example of a malicious command that could significantly impact the power system is the opening of the circuit breaker connected to

a generating power plant. The next section describes three specific use case scenarios through which a legitimate but unwanted/bad command can be injected into the cyber-physical power system.

3.4.1. Use Case 1: Adversary Impersonates Network

Use Case 1 Description: ***The adversary impersonates the network and sends a false (unwanted) but legitimate command outside of the control center to the circuit breaker of the largest generator.***

Communication Network under Attack: The communication network enables the following processes following the attack:

I. The Intrusion Detection System (IDS) notifies the control center operator of what command it has received. The operator verifies whether the command is legitimate

II. A false command is then issued to the substation RTU connected to the breaker of the targeted generator.

Power System under Attack: If the attack is successful, we can observe the following impacts on the power system:

I. An insecure operational state of the power system,

II. Possible shedding of electrical load

Use Case Steps:

1) The attacker sends a false but legitimate command from a location other than the control center to the generator breaker over an insecure network.

2) The IDS detects a suspicious, malicious command based on its rules engine based on IP

Addresses and port numbers and then notifies the operator. The operator verifies that the control center did not issue the command.

3) If the command was allowed to go through, CPSA evaluates the effect of the command on the power system and discovers that the system is insecure, indicating that the command was malicious.

4) The operator discards the command. Secure system operation is restored.

3.4.2. Use Case 2: Adversary Fabricates Legitimate Command

Use Case 2 Description: ***Adversary fabricates or modifies a legitimate command sent to a generator breaker over an insecure network.***

Communication Network under Attack: Same as in Use Case 1, except that the legitimate command was modified during transmission over the network.

Power System under Attack: Same as in Use Case 1.

Use Case Steps:

1) The adversary modifies a legitimate command transmitted from the control center to the generator breaker over an insecure network.
2) The IDS does not detect the command modification, but still sends a notification to the operator. The operator verifies that the control center did not issue the command.
3) Same as Use Case 1, step 3.
4) CPSA asks IT personnel for attack information with a response that there is suspicion of a MITM attack.
5) Same as Use Case 1, step 4.

3.4.3. Use Case 3: Adversary as an insider attacker

***Use Case 3 Description*: Adversary as an insider attacker (other person) at the control center sends a legitimate but unwanted command to the generator breaker.**

Communication Network under Attack: The operator receives a command notification from the IDS and finds the transmitted legitimate command was not issued by him/her. In the worst-case scenario, the operator ignores the notification and allows the execution of the command on the power system.

Power System under Attack: Same as in Use Case 1.

Use Case Steps:

1) The adversary acts as an insider attacker who has access privileges for sending a legitimate command to the generator breaker.
2) The IDS does not detect the insider attack and notifies the operator that it is a legitimate command. The operator verifies that the received command is the same as what was issued from the control center.
3) The generator breaker receives a false trip command and trips.
4) CPSA determines that the system is insecure, indicating that the command was legitimate but false (unwanted).
5) CPSA asks the IT personnel for attack information, receives the response that there is suspicion of an insider attack, and prompts the operator to reclose the breaker.
6) If the breaker does not respond within 20 seconds, CPSA will prompt the operator to initiate the appropriate remedial action, after which secure system operation is restored.

3.5. Challenges

Several challenges exist in modeling bulk electric systems (BES) cyber-physical security. During a cyber-physical attack such as a bad command injection, the power system may undergo various state transitions and eventually become insecure. Following are three specific challenges that make investigating cyber-physical system security difficult:

I. Scalability: due to the involvement of a large number of power and cyber devices

II. Limitations of the simulation tool: existing simulation tools are purpose-specific for power and communications networks, not designed for integrated operation.

III. Modeling the behavior of the physical system as impacted from cyberspace is not straight forward and requires expertise in both domains.

3.6. Cyber-Physical System Topology

The cyber-physical test systems developed for this dissertation consist of two layers – the communications network layer and the bulk electric system layer. The communication network topological structure of the cyber-physical system is shown in Figure 10. The communications network consists of a single control center with an EMS, where a power systems operator makes operation and control decisions, and 24 substations. The bulk electric system (BES) is shown in Figure 11. The communications network topology shown in Figure 10 was integrated with the BES in Figure 11 to form the cyber-physical BES test system shown in Figure 12. Test power systems were developed using the PowerWorld simulator as shown in Figure 12. Three different cyber-physical power systems were designed and tested using the same architecture. A table data structure was developed to hold the power system component parameters as shown in Figure 19. These parameters are used to re-initialize the power system after every simulation run. Table 9

captures the communication network parameters set during the co-simulation runs, while Figure 19 shows the network communication log developed to include the routing table and events for the communications between the RTU and control center.

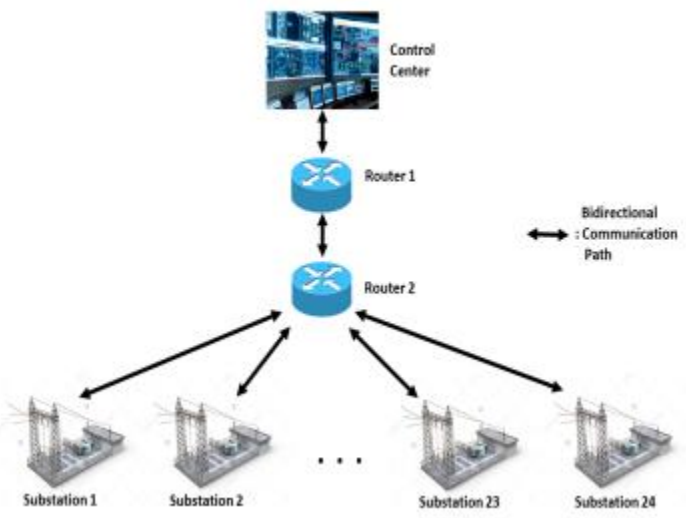

Figure 10- Communication network topology consisting of a single control center and multiple substations, each consists of an RTU.

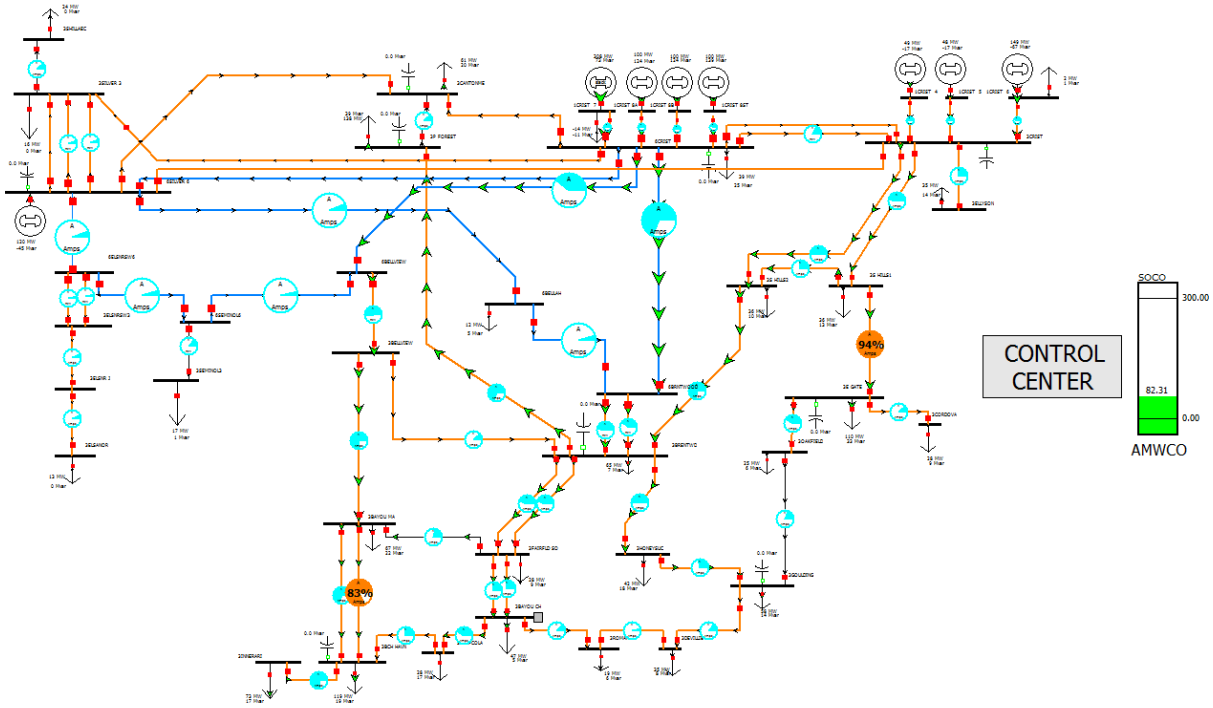

Figure 11 - Bulk Electric System Topology

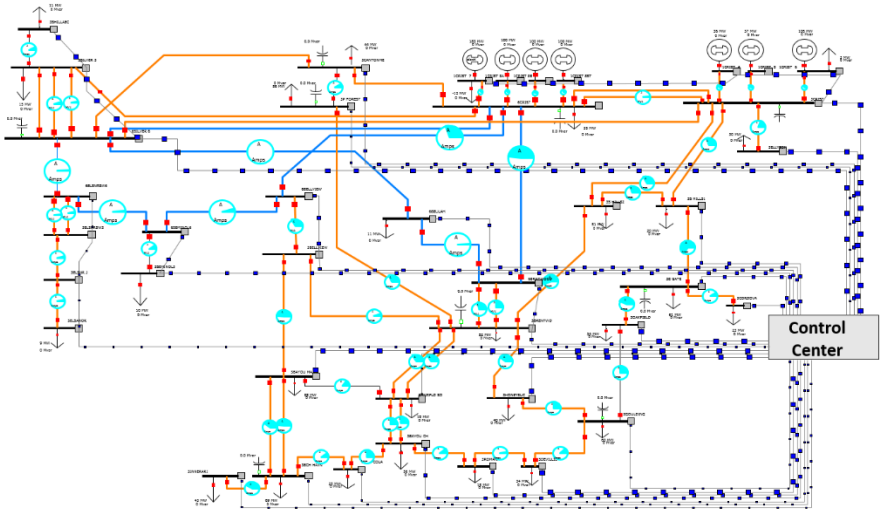

Figure 12- Cyber-Physical System with 24 substations

BRANCH	Tue 2016.05.03 at 03:17:25 PM EDT				
BusNum	BusNum:1	Line Circuit	Line Status	LineMW	LineMVR
1	7	1	Closed	-21.04	-0.37
2	3	1	Closed	10.20337	-0.14802
5	2	1	Closed	5.13083	-0.05707
5	2	2	Closed	10.20168	-0.05522
3	4	1	Closed	10.20168	-0.00341
5	6	1	Closed	8.832336	2.949891

Figure 13 - Sample Metadata for Bulk Electric System Branches

GEN	Tue 2016.05.03 at 03:17:25 PM EDT				
BusNum	GenID	GenStatus	GenMW	GenMWR	GenVoltSet
10	4	Closed	49.85	-22.3874	1
11	5	Closed	48.2	-22.3874	1
12	6	Closed	149.43	-86.7907	1
13	7	Closed	207.021	24.43159	1.0348
14	8	Closed	100	138.7	1.0348
15	8A	Closed	100	123.5	1.0348

Figure 14 - Sample Metadata for Bulk Electric System Generators

BUS	Tue 2016.05.03 at 03:17:25 PM EDT		
BusNum	BusName	BusPUVolt	BusRad
1	3SHILLAEC	1.014889	0.538019
2	3ELSNRSW	1.016529	0.542856
3	3ELSNR J	1.016344	0.541792
4	3ELSANOR	1.016179	0.540943
5	6ELSNRSW	1.016559	0.546357
6	6SILVER 6	1.015285	0.544642

Figure 15 - Sample Metadata for Bulk Electric System Buses

TRANSFORMER	Tue 2016.05.03 at 03:17:25 PM EDT			
BusNum	BusNum:1	LineCircuit	LineStatus	LineTap
5	2	1	Closed	1
5	2	2	Closed	1
6	7	1	Closed	1
6	7	2	Closed	1
8	9	1	Closed	1
28	10	1	Closed	1

Figure 16 - Sample Metadata for Bulk Electric System Transformers

LOAD	Tue 2016.05.03 at 03:17:25 PM EDT			
BusNum	LoadID	LoadStatus	LoadMW	LoadMVR
1	A1	Closed	21.04	0.37
4	A1	Closed	10.19974	0.112085
7	1	Closed	15.15717	0.402403
9	1	Closed	13.34872	0.513412
12	E6	Closed	1.856063	1/187481
13	EC	Closed	-12.1796	-9.14255

Figure 17 - Sample Metadata for Bulk Electric System Loads

SHUNT	Tue 2016.05.03 at 03:17:25 PM EDT	
BusNum	ShuntID	SSStatus
6	1	Open
21	1	Open
23	1	Open
24	1	Open
27	1	Open
28	1	Open

Figure 18 - Sample Metadata for Bulk Electric System Shunts

Each substation contains an RTU by design. The communication between the RTUs and the control center is provided through two routers (Router 1 and Router 2). In Figure 12, the blue lines are the communication links used for data transport between the control center and the substations, and the orange lines are the transmission lines.

Table 9 - Communication Network Parameters

Parameters	Range Value	Unit
Baud Rate	100 – 9600	Bits/s
Propagation Delay	10-500	Ms
Packet Size	50 – 500	Bytes
Number of Packets	1-5	-

2.0 receive router ad from	Router2										
5.3											
5.3 receive incoming	Packet#1	out of	1	with id	997260727	from	Output_CC_port1	to	RTU_1	tag	GridSimTags.delay 0
5.3 enqueing	Packet#1	out of	1	with id	997260727	from	Output_CC_port1	to	RTU_1	tag	GridSimTags.FLOW_SUBMIT
5.3 dequeuing	Packet#1	out of	1	with id	997260727	from	Output_CC_port1	to	RTU_1	tag	GridSimTags.FLOW_SUBMIT
10.3											
10.3 receive incoming	Packet#1	out of	1	with id	1721393242	from	Output_CC_port2	to	RTU_2	tag	GridSimTags.delay 0
10..3 enqueing	Packet#1	out of	1	with id	1721393242	from	Output_CC_port2	to	RTU_2	tag	GridSimTags.FLOW_SUBMIT
10..3 dequeuing	Packet#1	out of	1	with id	1721393242	from	Output_CC_port2	to	RTU_2	tag	GridSimTags.FLOW_SUBMIT
15.3											
15.3 receive incoming	Packet#1	out of	1	with id	339570773	from	Output_CC_port3	to	RTU_3	tag	GridSimTags.delay 0
15.3 enqueing	Packet#1	out of	1	with id	339570773	from	Output_CC_port3	to	RTU_3	tag	GridSimTags.FLOW_SUBMIT
15.3 dequeuing	Packet#1	out of	1	with id	339570773	from	Output_CC_port3	to	RTU_3	tag	GridSimTags.FLOW_SUBMIT

Figure 19 - Example Communication Event Log at Routers

Each substation is equipped with many power and field devices, including circuit breakers for connecting and disconnecting power components, transmission lines for carrying power from

generators to load centers, buses for connecting generators to loads, generators for producing power energy, loads for consuming required power demand, transformers for stepping up/down the voltage as needed, and shunt capacitor banks for maintaining the power system voltage. The control center polls each substation's RTU to send its available measurement data. All polled RTUs send their data back to the control center as a response. In practice, this communication sometimes takes place over encrypted communication channels dedicated to a utility's substation data transport. However, some internal substation communication networks may be insecure and unencrypted, and as a result, it is likely that an adversary can modify the measurement data before it leaves the substation and/or send malicious commands to breakers from the substation.

3.7. Cyber-Physical Security Attack Framework

This section presents the attack framework developed for modeling a bad command injection. Traditionally, operational real-time security assessment of BES involves only contingency analysis and worst-case scenario modeling, as shown in Figure 20, without considering contingencies in the cyber-layer.

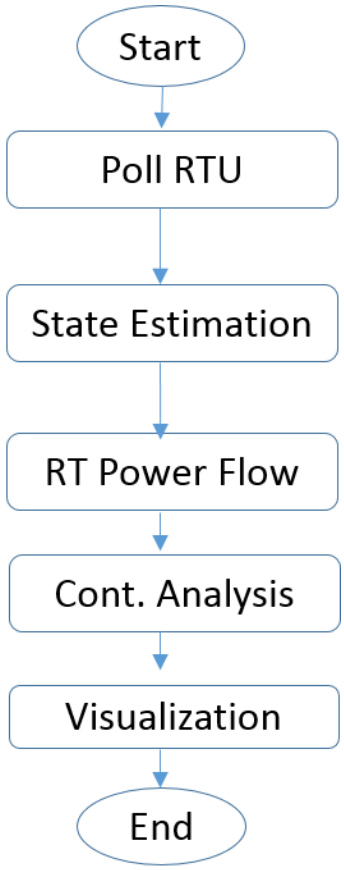

Figure 20 - Conventional Operational Security Assessment

However, the developed CPSA framework bridges the gap in security analysis by adopting a cyber-physical security approach to security assessment. The framework allows for the execution of a malicious command injection attack and assessment of the impact of the command execution on the power system by using the CPSA co-simulator, as shown in Figure 21. An adversary sends a malicious command encapsulated in a DNP3 packet from the control center (as an insider attacker) or from any other location (with a different IP address or spoofed IP pretending to be a legitimate IP address of the control center) to the substation RTU, which has specific IP and port number addresses.

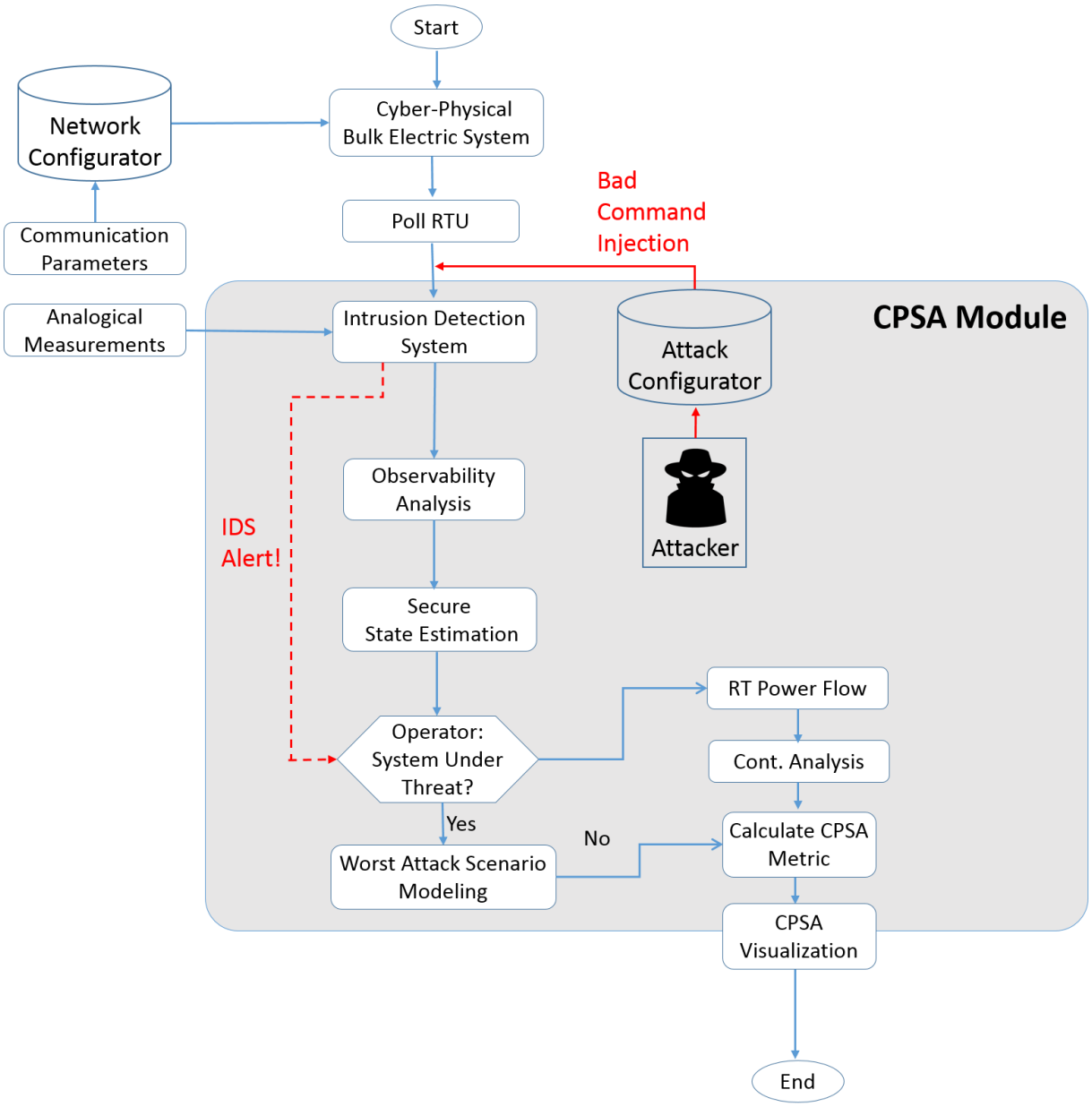

Figure 21 - Cyber-Physical Security Analysis

In this scenario, the IDS notifies the operator of the malicious command and instructs the operator to confirm whether that command is legitimate or not. Once the operator receives the malicious command attack information, it uses the co-simulator to model the malicious command injection in near real-time and decides whether to execute the command on the actual system. Descriptions

of the various components of the attack framework and their interdependencies are discussed in the following subsections.

3.7.1. WLS State Estimation Problem Formulation

State estimation is a data processing algorithm for converting redundant meter readings and other available information into an estimate of the state of an electric power system. The weighted least squares state estimator is implemented in the co-simulator. The formulation is presented here. Let z represent a set of power system measurements. Then, $z = h(x) + e$, where x is the estimated state vector (bus voltages and angles), h is the vector of functions relating the state variables to the error-free measurements, and e is a vector of measurement errors, which are assumed to have a Gaussian distribution with mean 0 and variance σ^2.

The WLS estimator minimizes the objective function

$$J(x) = [z - h(x)]^T R^{-1} [z - h(x)] \tag{4.1}$$

where R is a diagonal matrix of the measurement error variances. To obtain the minimum x, we take the partial derivative of the objective function

$$g\left(x^{(k)}\right) = \frac{\partial J(x)}{\partial x} = -H\left(x^{(k)}\right)^T R^{-1} \left(z - h\left(x^{(k)}\right)\right) \tag{4.1}$$

and set it equal to 0. Here, $x^{(k)}$ is the state vector at iteration k. H is the measurement Jacobian equal to $\frac{\partial h(x)}{\partial x}$.

By applying the Gauss-Newton method [96], we obtain the Normal Equations

$$\left[G\left(x^{(k)}\right)\right] \Delta x^{(k+1)} = -g\left(x^{(k)}\right) \tag{4.2}$$

where the gain matrix G is

$$G\left(x^{(k)}\right) = \frac{\partial g\left(x^{(k)}\right)}{\partial x} = H\left(x^{(k)}\right)^T R^{-1} H\left(x^{(k)}\right) \tag{3.4}$$

Then, we iteratively solve for x until a convergence tolerance ε is reached.

3.7.2. System Observability Analysis

The state estimator's performance relies on the quality, availability, and integrity of measurements, which in turn relies on substation sensing and communications devices. Observability analysis checks whether the availability of measurements is enough to estimate the power system state. If there are enough measurements to estimate the power system state, the system is considered observable [96]. A power system is said to be fully observable if the voltage phasors at all system buses can be uniquely estimated with the use of the available measurements. If the system is found not to be observable, this will suggest that there are unobservable branches whose power flows cannot be determined. Unobservable branches connect observable islands of an unobservable system. Typically, observability is done via two methods: topological analysis and numerical analysis. Here, we present the formulation for the numerical observability analysis used in the implementation of the cyber-physical co-simulator. Using the decoupled linear measurement model

$$H_{AA}\theta = z_A \tag{4.5}$$

the WLS estimate for θ will be given by

$$\hat{\theta} = (H_{AA}^T H_{AA})^{-1} H_{AA}^T z_A \tag{4.6}$$

If $H_{AA}\hat{\theta} = 0$ and $P_b = C\hat{\theta} \neq 0$, then $\hat{\theta}$ will be an unobservable state, where P_b , the vector of branch flows, is obtained from the dc power flow, and C is the reduced branch-bus incidence matrix. If $P_b(i) \neq 0$ for a branch i, then i will be called an unobservable branch. Below are the steps for implementing the numerical observability analysis algorithm:

1. Remove all irrelevant branches.
2. Form the decoupled linearized gain matrix for the $P - \theta$ estimation problem: $G_{AA} = H_{AA}^T R_A^{-1} H_{AA}$.
3. If G_{AA} is nonsingular, the system is declared fully observable; otherwise, the unobservable branches will be found as described above.
4. Remove the unobservable branches and all injections that are incident at the unobservable branches.
5. Go to Step 1

3.7.3. System Contingency Analysis

Contingency Analysis (CA) simulates the outage for a given set of power grid elements (typically transmission lines) under a given condition and evaluates the consequent events following the outage [97], [98]. CA methods assess the redundancy of power grids in the event of outages of some of the components and conclude that a grid is *N - k* secure if it can remain within a stable operation domain when *k* components are put in an outage situation. After the DC power flow was performed, and the Power Transfer Distribution Factors (PTDFs) and Line Outage Distribution Factors (LODF) were computed, contingency analysis was implemented in the co-simulator:

obtain DC single solution for base case

compute PTDF's, LODF's

for each contingency **do**

compute post contingency flows

report limit violations

end

3.8. Key Assumptions

Following are the key assumptions of the attack framework:

***System Susceptibility*:** The adversary can perform suspicious and/or malicious activities, including attempts to access the login credentials of the operator and/or device, transmission of fake/bad commands, such as opening a circuit breaker connected to a substation device, and disturbing network communications and packet data. The adversary will attempt to discover and exploit these susceptibilities in order to compromise (modify, control, or steal) critical power system infrastructure, information, and operations.

***Adversary's Capability*:** The adversary is capable of performing a MITM attack by altering or replacing a legitimate command, injecting a malicious command, or accessing the control center to send a legitimate command as an insider attacker.

Adversary's Accessibility: The adversary has knowledge of the communication network topology as well as the power system topology. The adversary also has enough resources to perform the required malicious or suspicious actions and has access to the system.

Critical Components: The primary goal of the adversary is to target critical power components, such as generators and transformers, in order to damage the power system or cause a service

interruption. The adversary could also target the routers to explore and access the communication system and exploit vulnerabilities. The goal of the power operator is to prevent adversary access to critical components by performing regular security analysis to detect any suspicious activity and deny illegitimate access.

***Key Assets*:** Key assets are the pieces of critical information that the adversary will seek. These key assets could be information about a regular schedule for polling operations at a specific substation or the IP address and other communications-related information for the control center. With these assets, the adversary can later spoof the control center and try to inject a malicious command. The objective is to identify this malicious activity and prevent such injections in the real power system.

Detection, Reaction, and Adaptation: In the worst-case scenario, the adversary successfully performs an undetected malicious action, such as injecting a malicious command. Our objective is to simulate the worst-case scenario and analyze the potential impact of the malicious command.

3.9. Cyber-Physical Security Metrics

The development of a cyber-physical security metric is imperative for CPSA analysis, cyber-physical attack countermeasures, and increased grid resiliency. Cyber-physical security metrics must combine a variety of domains. In the physical bulk electric system domain, metrics can be evaluated based on the impact of cyber-physical attacks on power flow and stability. In the communication network domain, the metric can incorporate parameters such as communication link failure and vulnerability path installation rates [79].

Several metrics have been developed to assess power grid cyber-physical security. A security-oriented stochastic risk management technique called CPIndex was executed on

individual host systems to dynamically capture and profile low-level system activities such as inter-process communications among operating power system assets, as demonstrated in [99].

To make cyber-physical security metrics more robust, some work has included humans in the loop. In [100], EliMet uses an automated inference technique based on a system administrator's responsive behavior to actively query operators regarding those states for which sufficient information was not gained during initial passive observation. It then uses the estimated security measure values for predictive situational awareness by ranking potential cyber-physical contingencies that the security administrator should plan for when operating the power system Security indices are used by utility operators to assess the status of power grids. The indices indicate whether the system will operate within an acceptable margin during contingencies or whether preventive actions are required to maintain system security. Calculating a security index requires developing an inner loop that should be iterated over credible contingencies. The inner iteration is then implemented for defined contingencies, and the security metric will be calculated and presented visually to inform system operators about the system status. A metric called the Aggregate Megawatt Contingency Overload (AMWCO) which is used to evaluate the system's security, is implemented in the co-simulator. The AMWCO determines measures and quantifies how much the transmission lines are overloaded as a result of a cyber-physical attack. The formulation for the AMWCO is presented here. First, the Aggregate Percentage Contingency Overload (APCO) is calculated as

$$APCO_{BRANCH\ jk} = \sum_{\substack{Contigencies\ that \\ overloaded\ branch\ jk}} (\%Overload - 100) \tag{4.7}$$

The AMWCO for each branch is then calculated by using the APCO asy

$$AMWCO_{BRANCH\ jk} = APCO_{BRANCH\ jk} * MVARating_{BRANCH\ jk} \quad (4.8)$$

The AMWCOs across all branches are summed up to give the system wide AMWCO:

$$SysAMWCO = \sum_{jk \in System} AMWCO_{BRANCH\ jk} \quad (4.9)$$

$$= \sum_{\substack{Contigency \\ Violations}} MWCO_{CONTVIOL}$$

3.10. Intrusion Detection System

The Intrusion Detection System (IDS) deployed at each substation scans every DNP3 packet sent and received by the substation RTU. The IDS also notifies the operator about each control command it receives and then requests verification. IDS implementation can be performed with the use of Suricata [101], which is an open-source implementation and provides rich functionality for a customized IDS system. Suricata evaluates functions on network messages and performs DNP3 deep packet inspection. The rules of the IDS are developed by using Domain Specific Language (DSL), which uses binary-valued functions. The IDS functionality (such as verifying read and write DNP3 commands) is modeled through JavaScript Object Notation (JSON), which provides specific classes, groups, and identifiers to represent the rules. The IDS scans the received DNP3 packet from the Distributed State Estimator (DSE) [102], triggers the specific rule based on filters applied to the packet, and passes it to the control center if the packet is not malicious. If the packet is suspected to be malicious, the IDS sends a notification (an alarm) to the control center. The IDS combines signature and behavioral analysis to protect the system

against known, unknown, and advanced threats. Detecting suspicious behavior involves several factors, such as measurement data threshold, protocol modifications, and tracking IP addresses and port numbers. In this research, we also perform a traffic analysis on received packets by using Wireshark with the jpcap/WinPcap tool. We also examine the communication patterns over several nodes (RTUs, control center ports, and routers). The process is carried out over an extended period of several days as opposed to micro-examination since the IDS alerts on specific protocol patterns tend to generate many false positives. To prevent data exfiltration, we ensure strict IT controls for both physical and cyber-security. We use data leak and loss prevention to expose the egress traffic carrying unauthorized critical information. We also impose strong policies for role-based and attribute-based access control, and encryption for securing last-mile communications [103].

3.11. Contributions and Conclusion

Section 3.7 presented the development of a new framework for assessing electric grid cyber-physical security, which makes the following contributions:

1. A heuristic for power system monitoring and control.
2. A methodology for accessing electric grid cyber-physical security in the case of contingencies caused by a bad command injection attack.
3. A metric for quantifying the impact of bad command injection attack.

The framework is robust in that in the case of an attack, and it flags a malicious command as suspicious by the IDS. Even if the IDS does not detect the malicious command, and the command is executed on the real system, this approach can detect the power system disturbance and report the effect to the operator, who can then take appropriate actions (such as sending other control

commands) to diminish the impact of the previously executed malicious command. This framework significantly enhances the cyber-physical security assessment capability of the operator at the control center compared to the capability of the conventional approach to security assessment.

CHAPTER IV

ATTACK PROPAGATION IN CYBER-PHYSICAL ELECTRIC GRIDS

Attacks such as bad data injection into electric grid measurements can cause disruptions that transcend the cyber realm and affect the physical world. This chapter introduces a graph-based attack propagation model that simulates a bad data injection attack and executes a heuristic defense strategy by using power system state estimation. The state estimator is used to identify maliciously injected data and adopt physical security metrics to determine attack mitigation actions. Visualization from the analysis performed by the simulation can guide the operator at the control center to take appropriate action to minimize disruption of the physical power system operation.

4.1. Existing Work

Many studies have investigated the vulnerability of power grids by developing threat and attack models as well as by simulating different attack scenarios in a controlled environment. However, important challenges remain. One critical challenge is how to develop reasonable approaches and models that can mimic cyber-physical attacks in reality. Such models need to incorporate both a communications network and the power system's layer. In the current literature, two of the major approaches for investigating the effects of cyber-physical attacks on power systems are bad data and bad command injection attacks [89]. In modeling the power system layer, researchers typically adopt (one of) three popular model categories pure topological models [104], pure power flow models [105], and hybrid models [106]. Each category has its own advantages and disadvantages. Another challenge that researchers may face is that attackers might have different knowledge of the cyber-physical power grids, such as power system topological structures, electric features, real-time information, communication network parameters,

transmission and listening ports, and access points. With different levels of knowledge, attackers may adopt different attack strategies.

Bad Data Injection (BDI) attacks have drawn wide attention in cyber-physical power grids and were first proposed by Liu and Ning [41]. Liu showed that attackers can manipulate field measurements and introduce bad data into certain state variables and bypass the existing techniques for bad measurement detection in power systems by exploiting their knowledge of the power system topology. Liu et al. proposed a bad data detection method based on adaptive partitioning state estimation. This estimation can raise the detection sensitivity by dividing the global power system into several subsystems. Bad data then can be located in a small area by multiple rounds of partitioning [107]. The authors in [52] analyze the cybersecurity of state estimators in SCADA systems operating in power grids by assuming that the attacker possesses only a perturbed model corresponding (only) to a partial model of the true system or even an outdated model. The attacker is then characterized by a set of objectives, and policies are proposed to synthesize stealthy deceptions attacks, both in the case of linear and nonlinear estimators. Real-time online detection of stealthy false data injection attacks in power system state estimation was explored in [108], which proposed an online anomaly detection algorithm that utilizes load forecasts, generation schedules, and synchro-phasor data to detect measurement anomalies and provide some insight into the factors that affect the performance of the proposed algorithm. The research/authors also proposed an empirical method to obtain the minimum attack magnitudes and the detection thresholds for meeting specified false positive and true positive rates. The work in [108] was extended in [109] by Li et al., who considered the sequential (online) detection of false data injection attacks in the electric grid. The authors aimed to manipulate the state estimation procedure by injecting malicious data into the monitoring meters. When unknown parameters in

the system, namely the state vector, inject malicious data, the set of attacked meters poses a significant challenge for designing a robust, computationally efficient, high-performance detector. Li et al. developed a sequential detector based on the generalized likelihood ratio to address this challenge.

To accurately understand the impact of bad data injection in cyber-physical power delivery systems, we need to go beyond just constructing undetectable bad data injection attacks and investigate the modeling and simulation of how these attacks may propagate through a cyber-physical power system. The goal of the work in this chapter is not to design undetectable bad data injection attacks or to develop better bad data detection tests, but rather the goal is to study how bad data injection attacks propagate in the system before they are identified and contained or mitigated by the operator.

4.2. Contribution

Our attack propagation model uses contextual information about the attacks by training multiple Markov Chain models. This chapter develops a novel framework that leverages attack graph notation and Markovian processes to demonstrate the propagation of malicious measurement injections into the power grid network, injections that are intended to make the System Operator issue incorrect but legitimate commands with disastrous consequences.

4.3. Attack Graph Semantics

We represent the attacker's strategy in two phases. The first phase is the preparation phase when the attacker needs to gather information and prepare all the necessary tools to execute an attack. Nodes 1-5 capture the preparation phase, as shown in Figure 22.

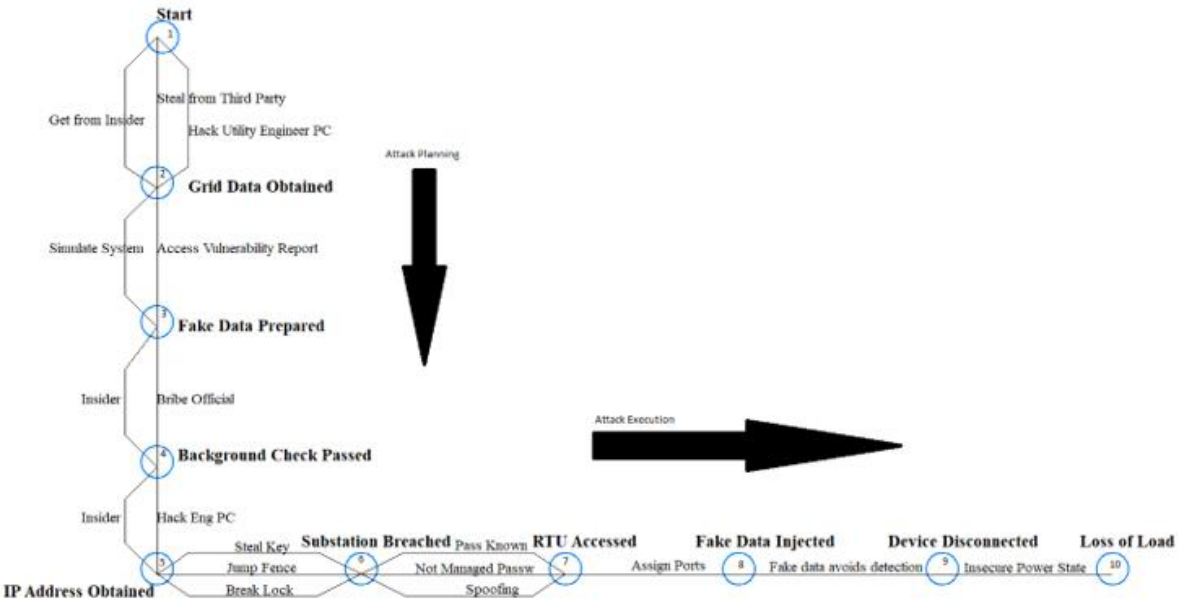

Figure 22 - Attack Graph Capturing Attacker's Strategy

The second phase is the execution phase when the attacker executes the attack and interacts with the defender. The success or failure of the attack propagating from the source to the operator at the control center is modeled by a Markov Chain, as shown in Figure 23, which encapsulates both the attacker's strategy and the probabilities of success/failure of the attack propagating from one node to the next.

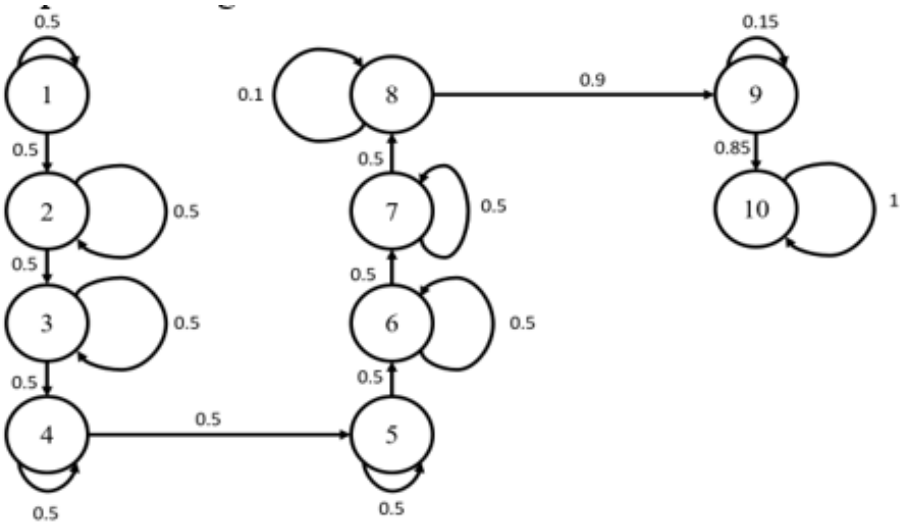

Figure 23 - Markov Chain capturing attacker's strategy for compromising the power system under attack assuming no defender

Each node represents a malicious task an attacker wants to complete in order to successfully reach their goal. In Figure 22, we assume that the tasks are sequential, which means that only one task is being executed at any point in time. We also assume that the system has bad data detection capability from the state estimator. The bad data detection of the state estimator detects and tries to prevent an ongoing attack from becoming successful; however, bad data detection does not reset the attacker's current progress. Therefore, successful detection of an attack at any node only pushes the attacker back to the immediately previous node. It is noteworthy to point out that the example scenario shown in Figure 22does not show the possibility that a successful detection by the defender may cause the attacker's progress to be set back by more than one node. The Markov Chain does not restrict how the edges are connected, so the edges do not need to connect with adjacent nodes. Each edge (directed) represents the attacker's attempt to complete the next task provided that the current task is completed.

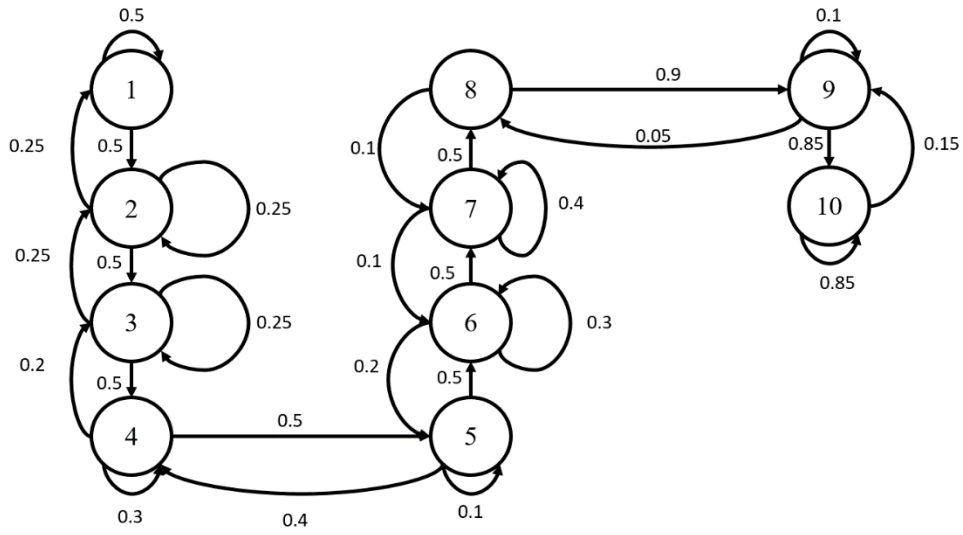

Figure 24 - Markov Chain capturing attacker's strategy for compromising the power system under attack assuming a defender with no state estimation

The values on the edges represent the probability that the attacker will successfully complete the next task. The probabilities used in Figure 23 - Figure 25 are informally assigned similarly to the assigning in [110]. The probability assigned to each edge only depends on each individual vulnerability, which is similar to many existing metrics, such as in the Common Vulnerability Scoring System (CVSS) [111]. As expected, there is a linear relationship between how fast the attack propagates and the probabilities on the edges relating to the attack. If the probabilities on the edge relating to defense are high, the probability of an attack going forward to the next edge decreases. Table 1 shows the mapping of an attacker's tasks to each node in Figure 22 - Figure 25. Figure 22 is the attack graph representation of the Markov Chains shown in Figure 23 - Figure 25.

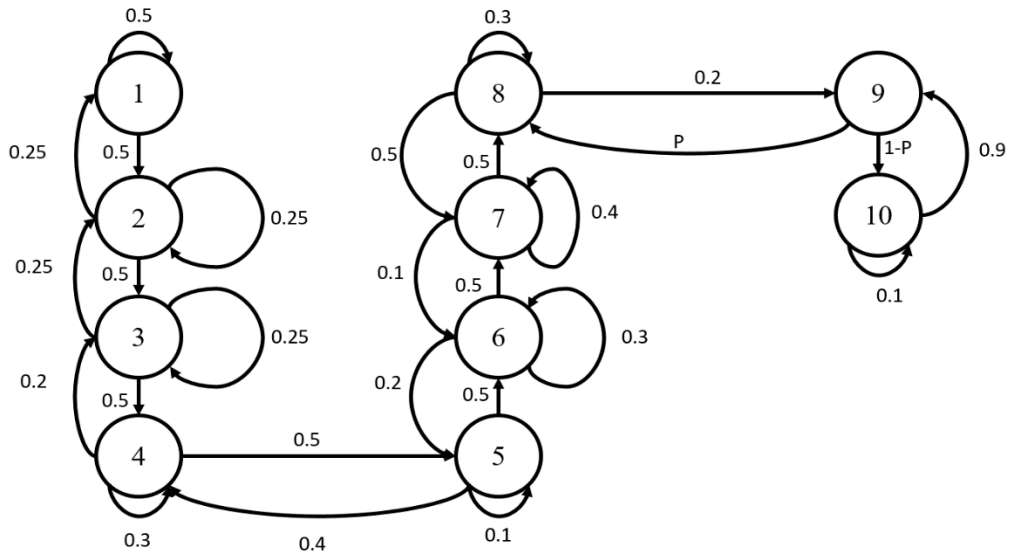

Figure 25 - Markov Chain capturing an attacker's strategy for compromising the power system under attack assuming a defender with state estimation

Table 10 - Mapping of Attacker's Tasks to Each Node

Node Number	Description
1	Start
2	Grid Data Obtained
3	Fake data Prepared
4	Background Check passed
5	IP Address Obtained
6	Substation Breached
7	RTU(s) Accessed
8	Fake Data Injected
9	Incorrect State
10	Loss of Load

4.4. Bad Data Injection Attack Scenario

We assume that the attacker has full knowledge of the power system topology and that in the planning state, it obtains IP addresses/TCP ports of the Monitoring System and the Remote Terminal Units (RTU) at the substations. The attacker breaks into the substation enclosure

containing the RTUs and communication radio, connects a malicious computing device to the RTU, and can access the RTUs. The attacker compromises the measurements polled by the Monitoring System by constructing packets containing fake field readings and then transmitting the readings to the Monitoring System, as shown in Figure 26 below.

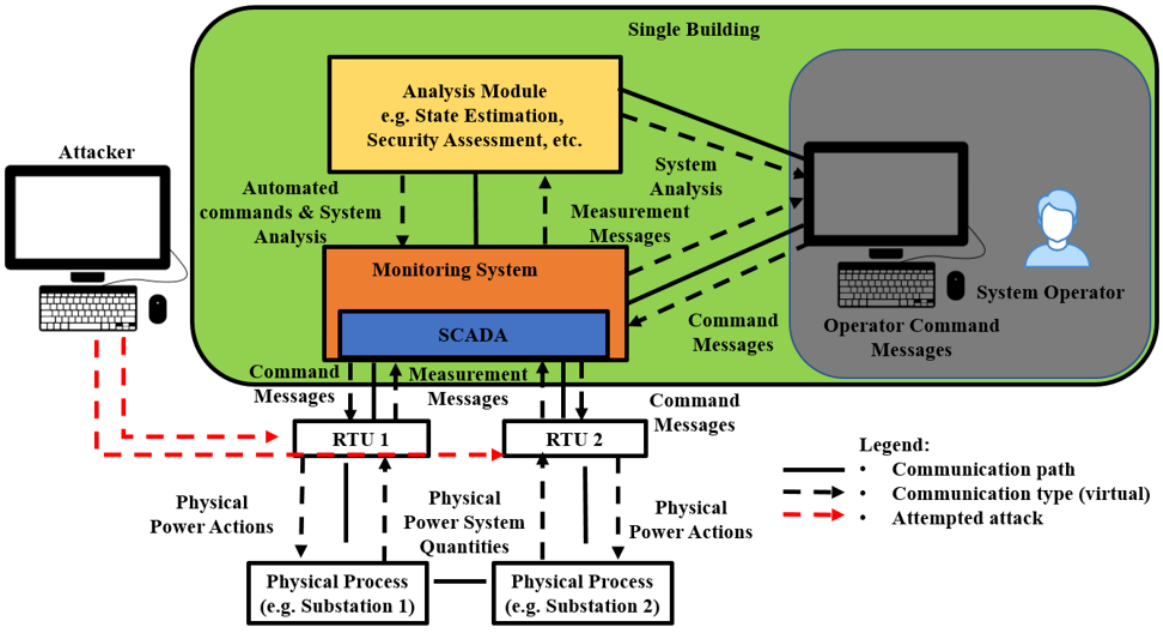

Figure 26 - Cyber-Physical System Bad Data Injection Attack Path

We also assume that the attacker's goal is to cause the System Operator to issue an incorrect command as a result of the fake data injection attack rather than to completely disable the RTUs.

4.5. Attack Propagation Model

The attack propagation model is a combined Markovian process and system state estimator model. As shown in Figure 22, we study attack propagation from the substation to the operator by

using a Markovian process. For a set of simulation time steps $t \in [1, T]$ the final state is obtained as shown in Equation 1. The x in Equation 1 is a vector representing the state of each node. Therefore, the x in our simulation is a vector of 10 elements, which represent the ten nodes in our attack graph. Each element in x is either "0" for uncompromised, or "1" for compromised. *P* is a matrix that contains probability values, as shown in Figures 2-4. The row number in *P* is the node number at the tail-end of the edge. The column number in P is the node number at the head-end of the edge. If there is no edge connecting two nodes, the corresponding value in the matrix is 0. An example *P* matrix which is used in Figure 3 is shown below.

$$\boldsymbol{P} = \begin{bmatrix} 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.25 & 0.25 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.25 & 0.25 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.2 & 0.3 & 0.5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.4 & 0.1 & 0.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.2 & 0.3 & 0.5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.1 & 0.4 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.1 & 0 & 0.9 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.05 & 0.1 & 0.85 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.15 & 0.85 \end{bmatrix}$$

$$\begin{aligned} \boldsymbol{x}^{(T)} &= \boldsymbol{x}^{(T-1)} * \boldsymbol{P} \\ &= \left(\boldsymbol{x}^{(T-2)} * \boldsymbol{P}\right) * \boldsymbol{P} \\ &\quad \vdots \\ &= \boldsymbol{x}^{(0)} * \boldsymbol{P}^{T} \end{aligned} \qquad (1)$$

The model incorporates both the Markov Chain that captures attack propagation and a state estimation with bad data detection capabilities. The bad data detection algorithm is implemented

at Node 9 of Figure 25. We investigate three cases – no defender, defender without state estimation, and defender with state estimation. In the attack graph shown in Figure 22, Nodes 1-5 represent the attacker's preparation process, which is described in Table 10. Nodes 6-10 represent the tasks executed by the attacker on to the power grid. Without a defender, the Markov Chain representation of the attack graph is shown in Figure 23. There is no edge that shows an attacker's attack is pushed back to the previous node. In Figure 24, the Markov Chain representation of the attack graph shows the existence of a defender, but one (a defender) without state estimation. Node 8 represents an attacker injecting fake data from the RTU. Node 9 represents the Analysis Module and Monitoring System in Figure 26. Without state estimation, there is a 0.9 (90%) chance that the fake data injected into the measurements will cause the Analysis Module to reach an incorrect system state, and then the Monitoring System displays the incorrect system state to the system operator. Furthermore, there is a 0.05 (5%) chance that the attack may be identified by the system operator, in which case the attacker's progress is pushed back to Node 8. Finally, there is also a 0.05 (5%) chance that the attack stays in Node 8 because the fake data has not reached the Analysis Module at Node 9. The probabilities are chosen to reflect the attacker's chances of successfully carrying out a bad data injection attack in the real world. Without a state estimator, the operator would have to rely on experience and intuition in order to analyze the massive amounts of data received at the control center, which means the chance of detecting bad data injection is low. In Figure 25, the Markov Chain representation of the attack graph assumes the existence of a defender with state estimation in Node 9. In this case, the state estimator can detect and eliminate the bad data, which enhances the detection capabilities of the power system. Therefore, the edge from Node 8 to Node 9 has the probability **P**. **P** is a variable probability of detection, which is calculated by the state estimator by using a/the Chi-square cumulative distribution function. The existence of

a state estimator provides a means to mitigate the attack by preventing the attack from propagating to the next node.

For simplicity, we model the bad data injection attack propagation by using a two-bus system with a generator and load, as shown in Figure 27. The field measurements from each bus are polled by two separate RTUs. The attacker compromises RTU1 measurements before compromising RTU2 measurements. RTU2 also acts as a Master Terminal Unit (MTU). An MTU can be an RTU that accepts different inputs such as field measurements from several RTUs and then transmits the measurements over the network to the analysis module for computational analysis.

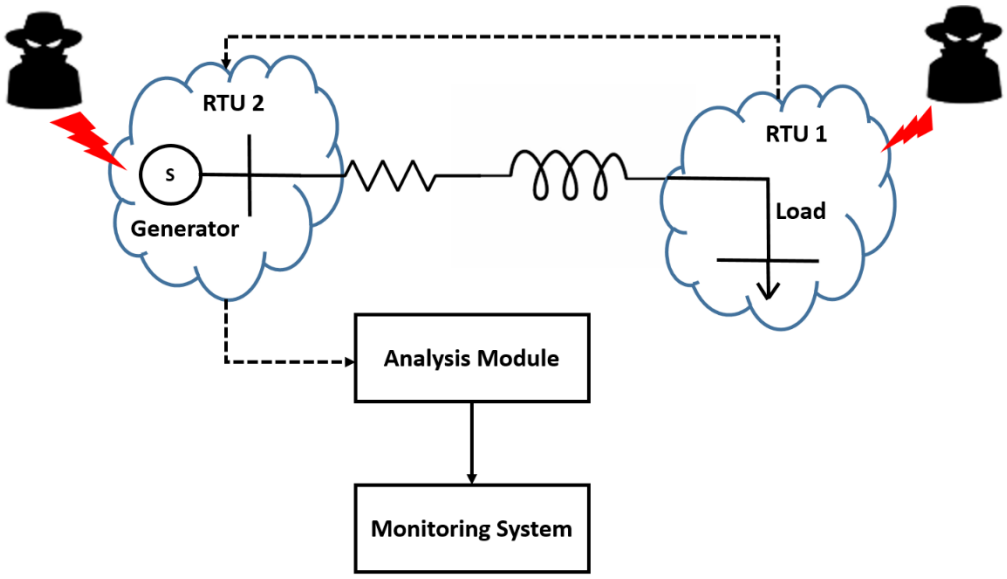

Figure 27 - Two-Bus Case under Bad Data Injection Attack

The monitoring system collates the measurements by concatenating the measurements into a single measurement vector, as shown in Equation 2.

$$\boldsymbol{z} = \left[\boldsymbol{z}_{\boldsymbol{RTU}_1}; \boldsymbol{z}_{\boldsymbol{RTU}_2}\right] \quad (2)$$

$$\boldsymbol{z} = \boldsymbol{h}(\boldsymbol{x}) + \boldsymbol{e} \quad (3)$$

$$\boldsymbol{z} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{e} \quad (4)$$

$$\hat{\boldsymbol{x}} = (\boldsymbol{H}^{\boldsymbol{T}}\boldsymbol{R}^{-\boldsymbol{1}}\boldsymbol{H})^{-\boldsymbol{1}}\boldsymbol{H}^{\boldsymbol{T}}\boldsymbol{R}^{-\boldsymbol{1}}\boldsymbol{z} \quad (5)$$

$$\boldsymbol{r} = \boldsymbol{z} - \boldsymbol{H}\hat{\boldsymbol{x}} \quad (6)$$

$$\left|\left|\boldsymbol{r} = \boldsymbol{z} - \boldsymbol{H}\hat{\boldsymbol{x}}\right|\right| \leq \boldsymbol{\tau} \quad (7)$$

$$\boldsymbol{z}_{\boldsymbol{attack}} = \boldsymbol{z} + \boldsymbol{a} \quad (8)$$

$$\left|\left|\boldsymbol{z}_{\boldsymbol{attack}} - \boldsymbol{H}\hat{\boldsymbol{x}}\right|\right| \leq \boldsymbol{\tau} \quad (9)$$

Using a standardized weighted least-squares state estimation model, Equation (3) shows how the various field measurements denoted by $\boldsymbol{z}$ are related to the state variables $\boldsymbol{x}$ (i.e., the voltages and phase angles) and the measurement error $\boldsymbol{e}$. $\boldsymbol{h}(\cdot)$ is a non-linear vector function expression of the measurements in terms of the state variables. Equation (4) shows the linear relationship between $\boldsymbol{z}$, and $\boldsymbol{x}$ under DC power flow model assumptions. Equation (5) denotes the state estimates, where $\boldsymbol{z}$ is the vector of measurements, **H** is the measurement Jacobian matrix, **R** is the error covariance matrix, $\boldsymbol{x}$, $\hat{\boldsymbol{x}}$ are the vectors of state variables and state estimates respectively, and $\boldsymbol{e}$ is the vector measurement error. The state estimates are considered valid only if the measurement residuals $\boldsymbol{r}$ are less than a threshold ($\boldsymbol{\tau}$), as shown in equations (6) and (7). The threshold is set based on state estimation residual information obtained from historical data when the system was operating normally. The attacker compromises measurements in the measurement vector $\boldsymbol{z}$ by changing measurement values, as shown in Equation (8), thus corrupting existing legitimate measurements.

For this simulation, the available measurements are taken to be

$$\boldsymbol{z} = \begin{bmatrix} V_1\,(kV) \\ V_2\,(kV) \\ P_{12}(MW) \\ Q_{12}(MVAr) \\ P_2(MW) \end{bmatrix}$$

and the quantities being estimated are $\boldsymbol{x} = [\theta_2, V_1, V_2]$. $RTU1$ measurements are relayed to $RTU2$, which collates those measurements with its own and sends the collated measurements over the backbone communication network to the monitoring center. A synthetic load profile with peak load at mid-day is adopted to drive the simulation. Equation (8) is adopted in the formulation for minimizing the weighted least squares state estimator objective function shown in Equation (10), where $\boldsymbol{h_i(x)}$ are components of the measurement Jacobian and R_{ii} is the diagonal matrix elements representing the standard deviation of each measurement i.

$$J(\boldsymbol{x}) = \sum_{i=1}^{m} \frac{\left[z_{attack,i} - h_i(\boldsymbol{x})\right]^2}{R_{ii}}$$

$$= [\boldsymbol{z_{attack}} - \boldsymbol{h}(\boldsymbol{x})]^T \boldsymbol{R^{-1}} [\boldsymbol{z_{attack}} - \boldsymbol{h}(\boldsymbol{x})] \qquad (10)$$

At the minimum, the first-order optimality conditions must be satisfied, thus requiring the following:

$$\boldsymbol{g(x)} = \frac{\partial J(\boldsymbol{x})}{\partial \boldsymbol{x}} = [\boldsymbol{H}(\boldsymbol{x})]^T \boldsymbol{R^{-1}} [\boldsymbol{z_{attack}} - \boldsymbol{h}(\boldsymbol{x})] \qquad (11)$$

where $\boldsymbol{H(x)} = \left[\frac{\partial h_i(x)}{\partial x_j}\right]$ is the measurement Jacobian. Expanding the nonlinear function $g(x)$ around a guess state vector $\boldsymbol{x}^k$ and dropping the higher order of terms leads to a Newton iterative solution:

$$\boldsymbol{x}^{k+1} = \boldsymbol{x}^k - [\boldsymbol{G}(\boldsymbol{x}^k)]^{-1} \boldsymbol{g}(\boldsymbol{x}^k) \qquad (12)$$

It is imperative to note that the estimated state of the system would be the compromised states that do not reflect the true state of the system because of the field measurements being compromised.

4.6. Simulation Results

As described in the previous sections, the attacker's strategy to compromise the power grid is broken down into multiple tasks that the attacker must complete in order to reach a certain outcome. In the example attack graph shown in Figure 22, the attacker's goal is to inject bad data into the power grid to fool the system operator into issuing an incorrect command, which can damage the power grid itself. The attacker manipulates field measurements to be telemetered from both RTUs to the Analysis module for the execution of important grid functions such as state estimation, as illustrated in Figure 27. To detect bad data, we use the Chi-squared distribution to identify the presence of bad data, followed by the largest normalized residual test that identifies the actual measurements to be removed. When the residual is higher than normal, the operator at the monitoring center is notified.

By using the Markov Chain equation (1), we are able to find the probability of bad data because an attack is present at a given node for a specific time step. We present three simulation scenarios: (1) no defender, (2) with a defender but no state estimation, and (3) with a defender and state estimation.

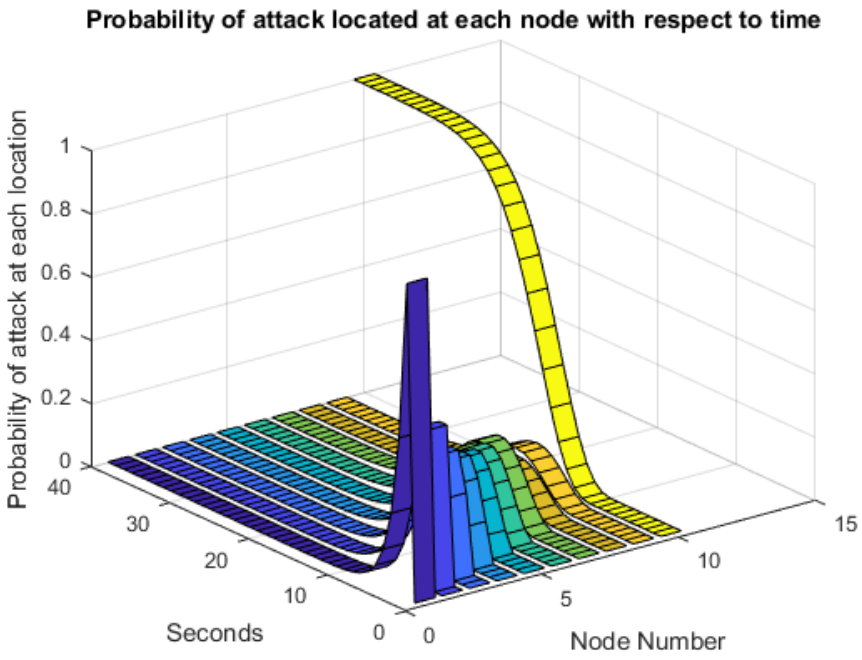

Figure 28 - Probability of an attack being located at each node with respect to time for the case assuming no defender

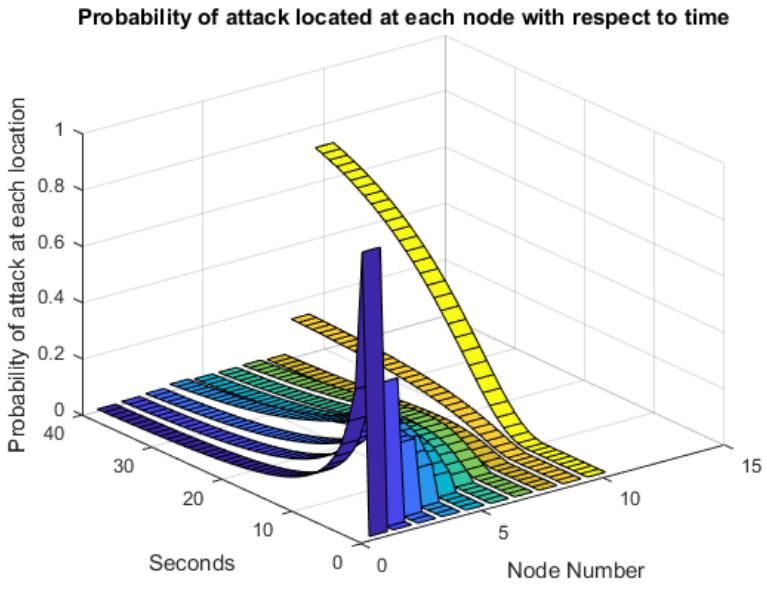

Figure 29 - Probability of an attack being located at each node with respect to time for the case assuming a defender exists, but not using state estimation

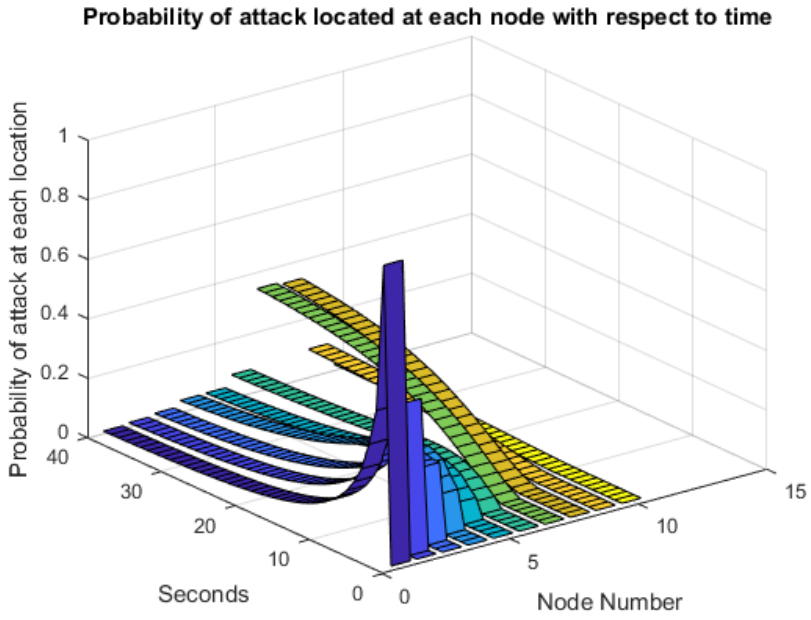

Figure 30 - Probability of an attack being located at each node with respect to time for the case assuming a defender exists and using state estimation.

All the simulation cases have the attacker starting their attack at a time equal to 2 seconds. In Figure 28, the simulation shows that the attack quickly propagates through Nodes 1-10, and so the probability of the attack being in Node 10 is 1 after only a few seconds. In this case, the attack propagates all the way through with the operator being presented with the incorrect state of the system, thus causing the operator to take incorrect action. In Figure 29, the simulation shows that the attack takes longer to reach Node 10, and after the simulation ends, the probability of an attack reaching Node 10 is less than 1. This means that with consideration of a defender in the power grid, the attacker does not always reach the goal with certainty. In Figure 30, the simulation shows that the attack still propagates through Nodes 1-5, but because of an increase in detection capability, the

probability that the attacker's attack stays at Nodes 7-8 is very high. In this case, the state estimation provides a viable means of increasing the defender's capabilities through the bad data detection functionality of the system state estimator. As a result, it is highly unlikely that the bad data propagates all the way to the control center, which could force the operator to decide based on inaccurate data.

4.7. Conclusion

The novel contribution of this research is the use of an attack graph to model attack propagation in a power grid scenario that combines the use of state estimation. Currently, the attack graph models only sequential tasks but does not model the parallel execution of an attacker's set of tasks. In addition, the attack graph does not consider either the attacker's ability to learn or the defender's ability to take away the knowledge gained by the attacker. The combination of the attack propagation model and state estimation provides additional information for the system operator so that appropriate mitigation strategies can be implemented. For example, if the simulation result shows that there is a high chance of an RTU being attacked, then the system operator may take actions such as password reset, disconnect the access point of an RTU, or increase the priority to send out substation crews to check on the compromised RTU. Results and insights from this work can inform the designing of better incidence response plans for operators at the control center. Operators can use the information provided from the visualization of the attack propagation and abnormal residuals to filter through information and pinpoint the most vulnerable sectors of the cyber-physical power system under attack and thus trip the necessary alarms and/or issue the appropriate control commands to contain the attack and mitigate its effects.

CHAPTER V

MODULAR DESIGN OF CPSA CO-SIMULATOR

This chapter discusses the modular design of the cyber-physical security assessment (CPSA) co-simulator. As shown in Figure 31, the design consists of various sub-modules interacting dynamically to model, capture, and analyze bulk electric systems containing multiple substations. The connected port of each substation as well as at the control center mirrors an Intrusion Detection System (IDS) for deep inspection of the packets flowing in the communication network. Below are the sub-modules of the CPSA tool:

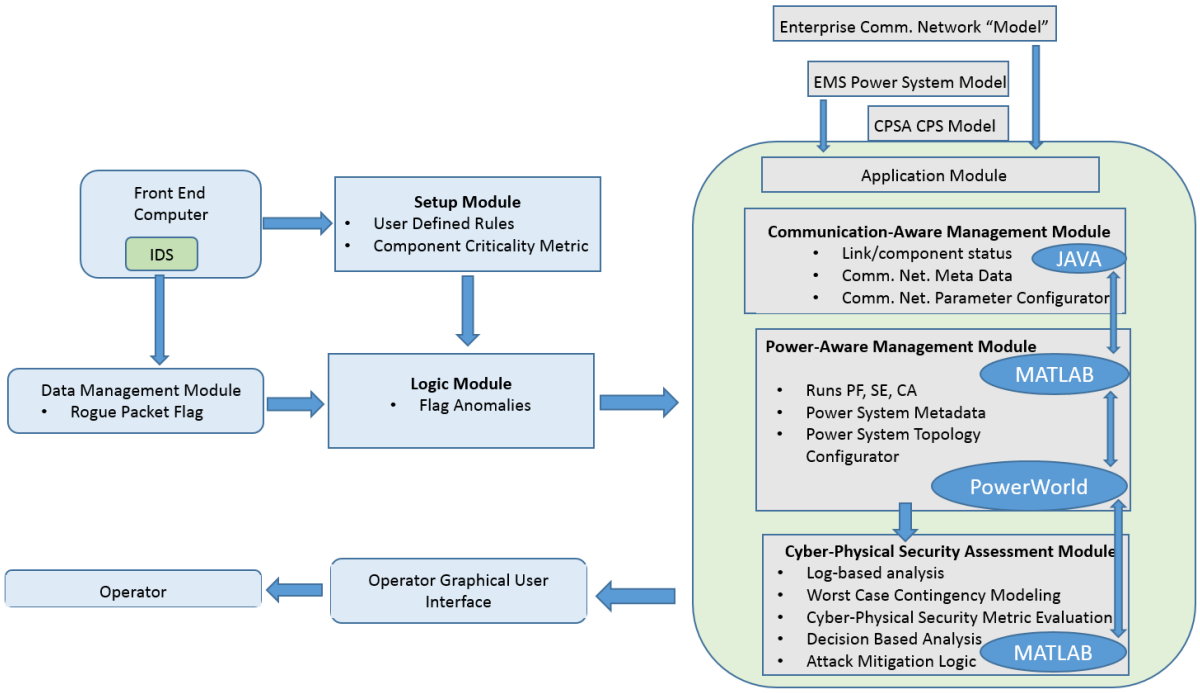

Figure 31 - Modular Architecture of Cyber-Physical Security Co-simulator

5.1. Data Management Module

The data management module stores all the measurement values, legitimate as well as rogue values, received in text files (extracted from the DNP3 packets). It stores rogue values with a flag “up" to distinguish them from legitimate data values. This module extracts measurement values from each packet or file and passes them to the next module, known as the logic module. The data management module uses buffer storage available at the control center for storing the packets.

5.2. Setup Module

The setup module specifies the user-defined rules, such as acceptable operational limits. It also provides a component-criticality metric, which clearly defines different components of the cyber-physical system by their severe criticality of loss.

5.3. Logic Module

The logic module verifies the boundary limits of each measurement value. If the module identifies bad measurement values, it separates out those values by setting the flag “up" for them but still passes the bad values in order to assess their effect on the power system under the bad measurement injection attack scenario to verify how much these values would impact the current state of the system (if they went undetected).

5.4. Cyber-Physical System Input Module

This module is comprised of the enterprise communication network model as well as the EMS power system model for the existing cyber-physical electricity system. The module implements the following models:

A. Enterprise Communication Network Model:

It provides input to the co-simulator regarding various communication components, including the communication network topology, the number of connected devices in the network, the baud rate, the packet size, the Maximum Transmission Unit (MTU) size, and the propagation delay over the communication channel.

B. EMS Power System Model:

It provides the co-simulator with power system input, which includes the power system topology, different parameters (with the actual value as well as acceptance ranges) for components such as transmission lines, buses, generators, loads, shunts, and transformers, and the configuration of the power system at the time of data acquisition.

5.5. Cyber-Physical System Application Module

This module is the main functional and application-driven module. It runs every few (4-5) minutes to check the current status of the system as it completes one cycle of 8-time steps in this duration. It performs all cyber-physical operations of the CPSA co-simulator and constitutes the main interface of the simulator, as shown in *Figure 32* below.

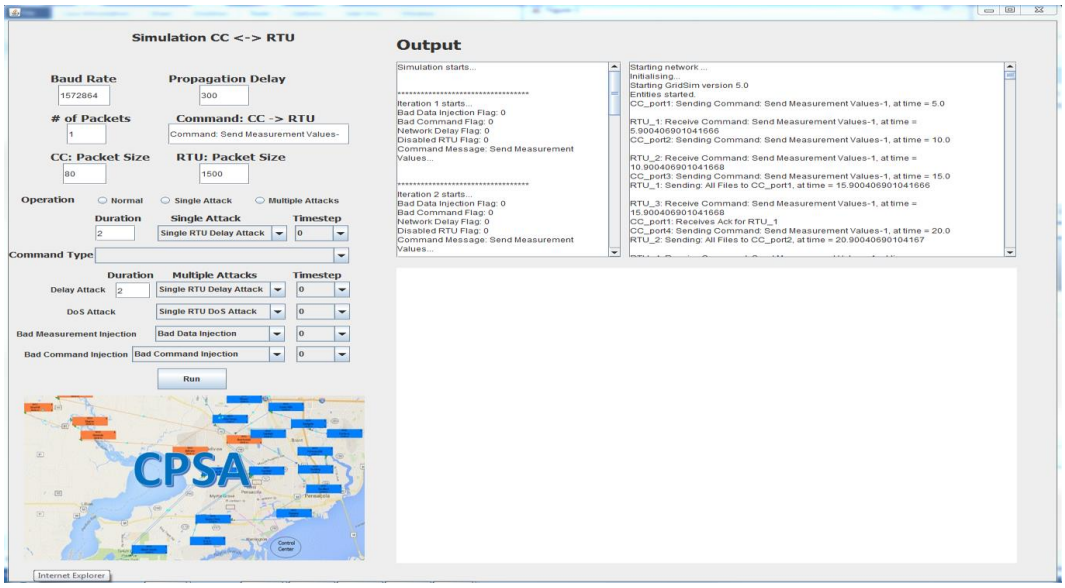

Figure 32 - Main Graphic User Interface of CPSA Co-Simulator

Based on the analysis and observations of this module, instructions for appropriate actions are forwarded to the security assessment module, which calculates the overall cyber-physical security metric for the system. This module performs the following sub-modular tasks:

A. **Communication-Aware Management Module:** It is responsible for managing different components of the communication system along with the statistics of any cyber-attack impact. Normal operations performed by this module include frequent pings to different communication devices to verify whether they are active and up and maintaining log records of the communications at the control center, RTUs, and intermediate devices, such as routers. This module consists of sub-modules described in more details thus:

I. **Communications between Different Components:** The co-simulation is made real-time by communication between the control center and RTUs through the provided routers,

where the sender can send multiple messages with a specified MTU size at one time, and the receiver responds with an acknowledgment for each message along with the action that needs to be performed, as shown in *Figure 33*. The communication system also includes a propagation delay and the delay at components for computations.

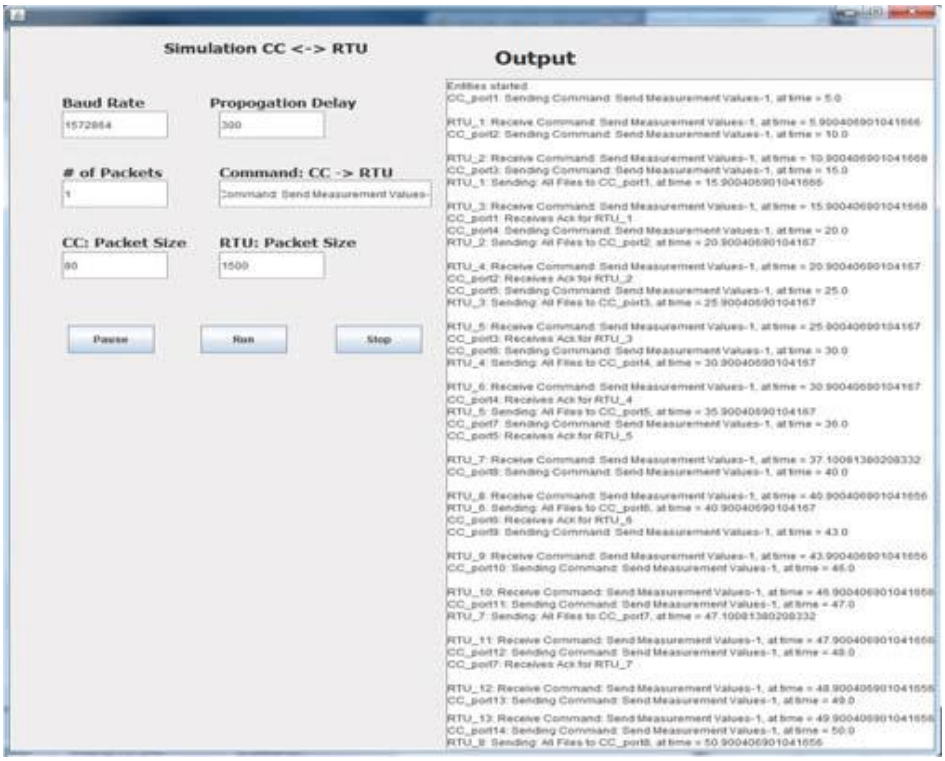

Figure 33 - A polling request initialed by the CC and RTUs reply with the current measurement values.

II. **Log Records of Communication Components**: The communication system maintains log records at the control center, at all RTUs, and at the routers. The logs include messages sent and received by the control center and RTUs, the enqueue and dequeue timing of each packet at each router along with sender and receiver information, and the route traversed by each message from the source to its destination.

0.0 Creates CC_port1									
5.0 CC_port1:Sending Command: Change Breaker/Line Status-1	At time 5.0								
17.100400901014167 CC_port1: Receives Ack for RTU_1									
CC_port1									
0.0 Creates RTU_1									
6.050403901041667 RTU_1: Receive Command: Change Breaker/Line Status-1	At time = 6.05040690101667								
16.05040690104167 RTU_1: Performing operation at time = 16.05040690104167									
1017.1004069010417 RTU_1:exiting									
RTU_1									
0.0 attach this ROUTER	to entity	RTU_1	Packet scheduler	RTU_Sched_1					
0.0 attach this ROUTER	With router	Router1	with link	R1_R2_link	packet scheduler	R2_Sched			
0.0 advertise to router	Router1								
2.0 receive router ad from	Router1								
5.75									
5.75 receive incoming	Packet #1	out of	1	with id	656534929	from	Output_CC_pot1	to	RTU_1
5.75 enqueing	Packet #1	out of	1	with id	656534929	from	Output_CC_pot1	to	RTU_1
5.75 dequeuing	Packet #1	out of	1	with id	656534929	from	Output_CC_pot1	to	RTU_1
16.3504069									
16.35040690104167 receiving incoming	Packet#1	out of	1	with id	710964375	from	Output_RTU_1	to	CC_port1
16.35040690104167 enqueing	Packet#1	out of	1	with id	710964375	from	Output_RTU_1	to	CC_port1
16.35040690104167 dequeuing	Packet#1	out of	1	with id	710964375	from	Output_RTU_1	to	CC_port1
1017.1004069010417 receives	Packet#1	signal							
	END_OF_SIMULATION								
ROUTER_2									

Figure 34 - Maintaining log records of the communication network statistics.

III. **Evaluate System Behavior under different Cyber-Attacks Scenarios:** The communication system is simulated in the presence of different cyber-attack scenarios so that the overall impact and the behavior of the cyber-physical system can be observed. The attack modeler, as shown in *Figure 35*. is used to simulate a bad command injection attack. A bad data injection attack affects the communication system components and results in the system behaving differently than it does in normal operation.

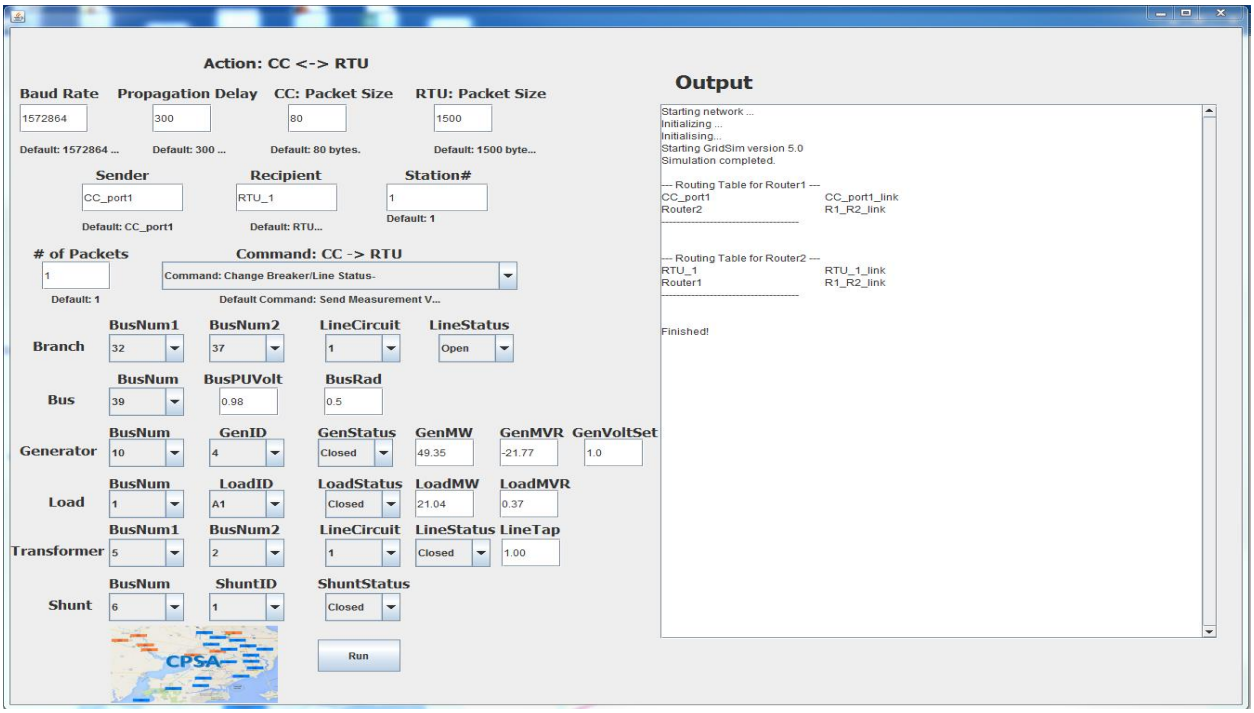

Figure 35 - Attack Modeler of CPSA Co-Simulator

IV. **Evaluate System Behavior with Future Demands Scenarios:** Based on future forecasts, such as the predicted load profile and generation dispatch (for example, the next 30 minutes), future states of the cyber-physical system are observable. This enables the co-simulator to run and evaluate system states faster than in real-time. After each co-simulator run of 2 minutes for 30 iterations, the system states for the next 30 minutes can be accurately predicted and analyzed.

B. **Power-Aware Management Module:** This module analyzes the current state of the power system by comparing legitimate and malicious or suspicious measurement values to evaluate their impact on the overall CPS security. It then simulates the what-if scenarios by using contingency analysis. It also verifies whether the suspicious measurements should be

forwarded to other applications if the system is still secure. This module contains enhanced versions of three core power system functions typically performed by the EMS: global state estimation, power flow, and contingency analysis.

5.6. Security Assessment Module

This module is specifically designed for operators to analyze the CPS system behavior based on a variety of observations provided by other modules. This module evaluates a trust metric to figure out the critical components of the cyber-physical system and performs a log-based analysis to verify secure operation. It can investigate any unexpected behavior it finds in any communication or power system component. Finally, the operator concludes with decision-based analysis and takes suitable actions in order to maintain the secure and stable operation of the power system.

5.7. Contribution and Conclusion

This chapter described the modular design of the co-simulator, which serves as the core for the software implementation presented in the next chapter. The following is an outline of the contributions of this chapter:

1. Modelled and implemented a worst-case contingency analysis that captures the cyber-physical security of a test bulk electric system.
2. Designed and implemented a heuristic for decision making for a test bulk electric system under a cyber-physical attack.
3. Implemented a heuristic for log-based analysis of bulk electric systems under a cyber-physical security attack.

CHAPTER VI

CYBER-PHYSICAL SECURITY ASSESSMENT CO-SIMULATION SOFTWARE FRAMEWORK

This chapter discusses the development of the software framework for the integrated time-driven, event-oriented, cyber-physical security co-simulator designed and implemented in this dissertation. In particular, it describes a co-simulation infrastructure that is required for the implementation of the cyber-physical security assessment application framework development. The software framework describes the software libraries developed or extended, while the application framework describes the different application modules implemented in the co-simulator to address the following requirements: (1) How to determine the physical impact of various types of cyber-attack on the operation of the grid, (2) How to quantify the cyber-physical security of a system in real-time, (3) How to increase operator situational awareness of system-level cyber-physical security through cyber-physical metrics and visualization.

6.1. Cyber-Physical System Co-Simulation Paradigms

The area of electric grid cyber-physical security co-simulation and testbeds have not been fully explored. In this direction, Davis et al. [112] presented a survey of cyber ranges and categorized these ranges as

1. Modeling and simulation, where models of each component exist
2. Ad-hoc or overlay, where tests are run on production network hardware with some level of test isolation provided by a software overlay, and
3. Emulation, which maps a desired experimental network topology and software configuration onto a physical infrastructure.

6.2. CPSA Co-Simulation Software Framework Implementation

To realize the capability to simulate power and cyber elements, we combined and extended multiple existing software modules in a co-simulation environment that enables the tracking of the execution sequence of each module. This *co-simulation framework* implements the necessary interfaces and achieves reasonable computational performance. The co-simulation framework is based on the combination of a GridSim smart grid simulator for modeling the *communications network, event scheduler, model validation module, data management module,* and *I/O module*. The power system is modeled in PowerWorld. The *computation module* is housed in MATLAB, where the cyber-physical application framework is implemented. The application framework runs algorithms such as state estimation, contingency analysis, etc. GridSim and MATLAB interact using TCP/IP sockets, with GridSim being the server and via a proxy server *MatlabControl.java with MATLAB* being the client. The proxy server handles all request and message transfers between GridSim and MATLAB. The TCP/IP connection enables running all software on a single computer or using a remote computer to run MATLAB and PowerWorld. The Instrument Control Toolbox is required for MATLAB to support TCP communication; however, any other toolbox enabling TCP connections could also be used.

On the other end of the co-simulation, MATLAB and PowerWorld are connected and interact through a Simulator Automation Server (SimAuto) that coordinates data exchanges and message transfers between instantiated COM objects. The interface between Matlab and PowerWorld/SimAuto described here is modified so that it can act as a gateway for requests issued by JADE agents. The MATLAB interface allows for the integration and interaction of the different simulators into a combined co-simulation framework, as shown in Figure 36

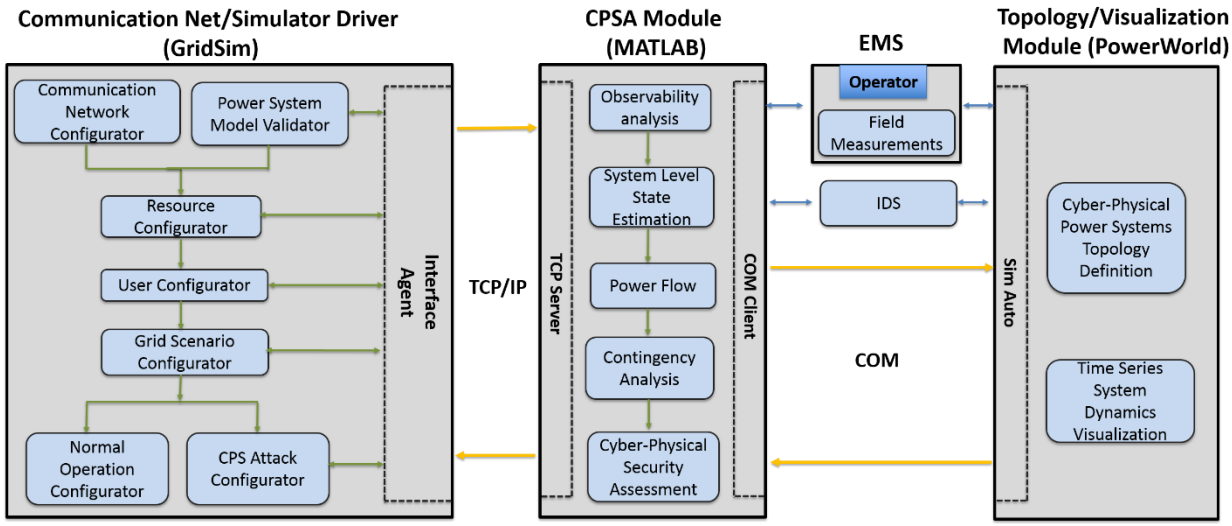

Figure 36 - Co-simulation Software Implementation Framework with Interfaces designed between GridSim (JADE), MATLAB and PowerWorld

Below is a brief description of the different platforms used in the development of the co-simulator.

A. **GridSim:** The GridSim toolkit allows modeling and simulation of entities in both parallel and distributed computing systems. It provides an environment for creating different classes of heterogeneous resources for solving computing and data-intensive applications. The processing nodes within a resource can be heterogeneous in terms of processing capability, configuration, and availability.

B. **MATLAB/Matlabcontrol:** Matlabcontrol is a Java API that makes it possible to call MATLAB from Java. It provides the ability to evaluate a variable (eval) and a function (feval) and allow *get* and *set* variables from Java to MATLAB.

C. **JADE:** JADE is used to provide an interface between the communication network (in Java) and the power system (in PowerWorld) through an interface by using MATLAB. JADE is an open-source middleware and a Java-based framework that facilitates the creation of agent-

based simulations by providing basic functionalities, such as agent and easily extended behavior classes. Although many other multi-agent frameworks are available, JADE is the most commonly used for power system applications.

D. **PowerWorld**: PowerWorld is a power systems simulator for simulating high voltage power systems. This tool allows/makes possible power flow analysis on a system with up to 100,000 buses. Multiple add-ons allow the performing of additional analysis such as transient stability, optimal power flow, voltage stability, reserves, transfer capacity, etc. SimAuto, which is an add-on used to control the simulator from external applications, acts as a COM object with which other software can communicate by sending requests and receiving data.

6.3. GridSim System Architecture

A multi-layer architecture and abstraction for the development of the GridSim platform and its applications are shown in Figure 37. The first layer of GridSim is concerned with the scalable Java interface and the runtime machinery - JVM (Java Virtual Machine). The JVM has been modified and extended for multiprocessor systems where each RTU and the control center that have been defined in the system run dedicated individual processors with assigned grid resources (instantiated in the third module). The second layer is a discrete-event infrastructure implementation built with the SimJava interface provided by the first layer. The third layer models and simulates core Grid entities such as resources, information services, uniform access interface, and primitive application modeling and framework for creating higher-level entities. We extended the GridSim toolkit [113] to focus on system entities modeled as RTUs, which use the discrete-event services offered by the lower-level infrastructure to respond to polls for data initiated by the control center. The last layer implements application and resource modeling with different

scenarios by using the services provided by the two lower-level modules in order to evaluate scheduling and resource management policies and algorithms. It is possible for the test case to become corrupted from previous co-simulation runs during the input/output (I/O) read, write, or initialization process. Hence, a model validation layer has been built into the application module. At the start of every co-simulation process, this module validates the power system model parameter against the original test case parameters stored in a local database. This is to ensure that the co-simulation is run on the test case with the correct parameters.

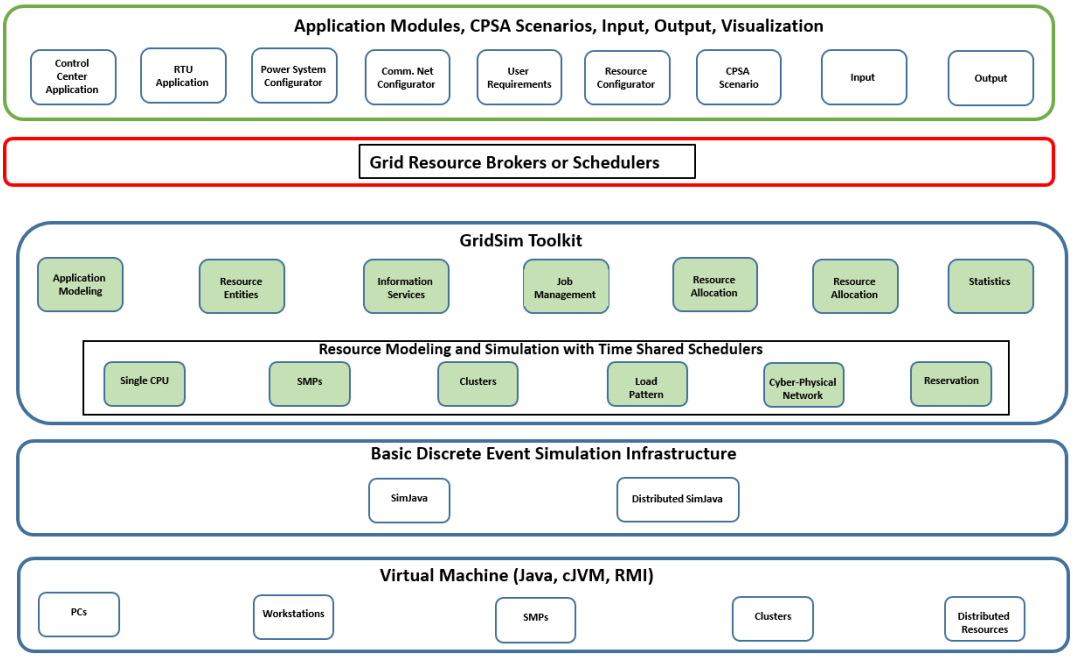

Figure 37 – An extended modular architecture for GridSim platform and components

Designing and implementing the communication network and the co-simulation driver required creating and/or extending GridSim entities as time-shared systems. Descriptions of these entities are presented below.

6.4. SimJava Discrete Event Model

The GridSim discrete event simulation is handled by SimJava, while a general-purpose discrete event simulation package is implemented in Java. The simulations in SimJava contain a number of entities modeled as RTUs and a Control Center. All entities were implemented to run in different parallel threads. We encode each RTU's behavior in Java by using its body( ) method, which requires implementing the following core primitives which were adapted from [113] and extended in this work:

- **sim schedule()** This method sends event objects to other entities via ports. It encapsulates commands issued from the control center and sends the commands to the RTU the command was intended for.
- **sim hold()** This executes holds for a specified simulation time and is implemented by all RTU's, leaving enough time for all polled data from the RTUs to be collated by the control center before the next stage of the simulation is initiated.
- **sim wait()** This allows either the RTU or control center to await an event object's arrival. The control center implements this method when waiting to receive all polled data from the RTUs in the field.

These methods and attributes facilitate a network of active entities that communicate by sending and receiving passive event objects efficiently. Because logging the arrival, departure, and execution of all commands and instructions is crucial in the co-simulation, a central object ***Sim_system*** creates a timestamped ordered queue of every event. The list of events is created from the user-specified scenario at the beginning of each simulation, at which time all entities are created and their **body()** methods are executed. Whenever any entity executes a simulation function, the ***Sim_system*** object halts that entity's thread and places an event on the future queue to indicate

execution of the function. When all entities have halted, ***Sim_system*** removes the next event off the queue, advances the simulation time accordingly, and restarts entities as appropriate. This process is dynamically repeated until no more events are generated.

6.4.1. GridSim Entities

The RTUs were designed with the capability for simulation as a single processor machine with heterogeneous resources that can be configured as time-shared systems. During the simulation, GridSim creates the specified number of multi-threaded RTUs and abstracts all the entities (control center and RTUs) and their time-dependent interactions in the real system. It supports the creation of user-defined time-dependent response functions for all interacting entities. The response function is both a function of the past and current states of the entities. The GridSim simulation contains entities for the control center application, RTU application, users, brokers, resources, statistics, and network-based I/O, as shown in the flow diagram in Figure 39. These entities were adapted from [113] and extended in this research work:

User. Each instance of the User entity represents a Grid user. A User entity is implemented as a Control Center or an RTU. The User entity creates a simulation instance or an experiment that contains an application description, which is a list of simulation actions to be processed. A network of active entities - RTUs and the CC - are instantiated and communicate by sending and receiving passive event objects efficiently. The control center User instance differs from the RTU instance with respect to the following characteristics:

Polling: Only the control center can execute polling commands.

Scheduling Optimization Strategy: Both the Control Center and the RTU User instances implement a time minimization strategy by immediately sending requested data once they received the request.

Broker. A Broker entity implements the round-robin algorithm and is responsible for scheduling tasks. At the start of every simulation, each user is dynamically bound to an instance of the Broker entity. Tasks initiated by a User are submitted to its associated broker, which then schedules the tasks according to the User's scheduling policy. The broker, using a get method, acquires a list of available resources from the global directory entity before scheduling the tasks. The broker then coordinates the execution of the tasks between the Control Center and the RTUs.

Resource. The Resource entity represents a Grid resource. Resources instantiated during the simulation differ from each other in the following aspects:

- number of processors
- speed of processing
- internal process scheduling policy – time shared or space-shared

The resource speed and the job execution time are defined in terms of the *Millions of Instructions per Second* (MIPS).

Grid information service. This provides information about the simulation setup, such as the number of RTUs and communication routers defined for a particular simulation run. It also performs resource registration services and keeps a record of resources available in the Grid. The

instantiated brokers can request configuration and status information from the Grid Information Service entity.

Input and output. Input and Output entities facilitate the flow of data between the Control Center and the RTUs. All GridSim entities instantiated during the simulation have assigned I/O ports for establishing links between other entities and their own Input and Output entities. The entity communication model via Input and Output entities in GridSim is depicted in Figure 38. Implementing separate threads for both the Input and Output entities allows for full-duplex message passing communications among different entities.

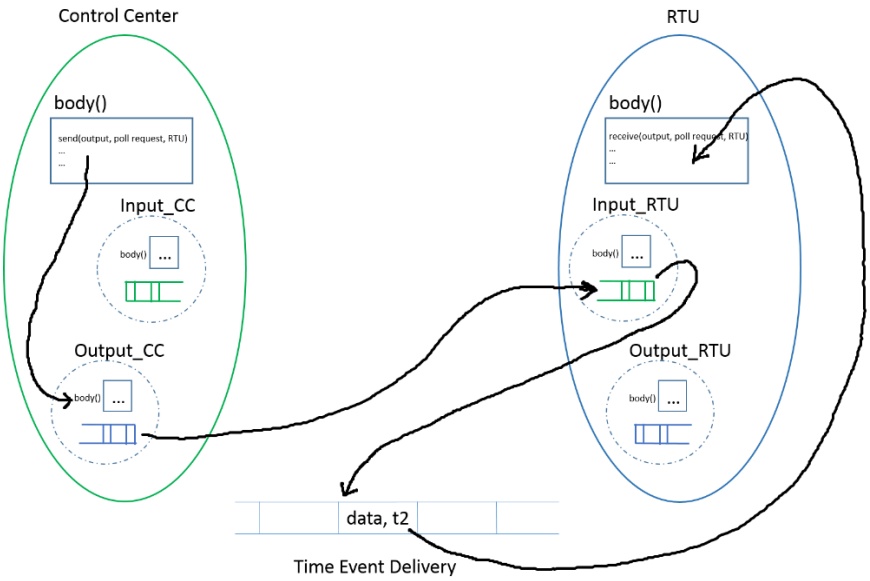

Figure 38 - Entity communication model via its Input and Output entities

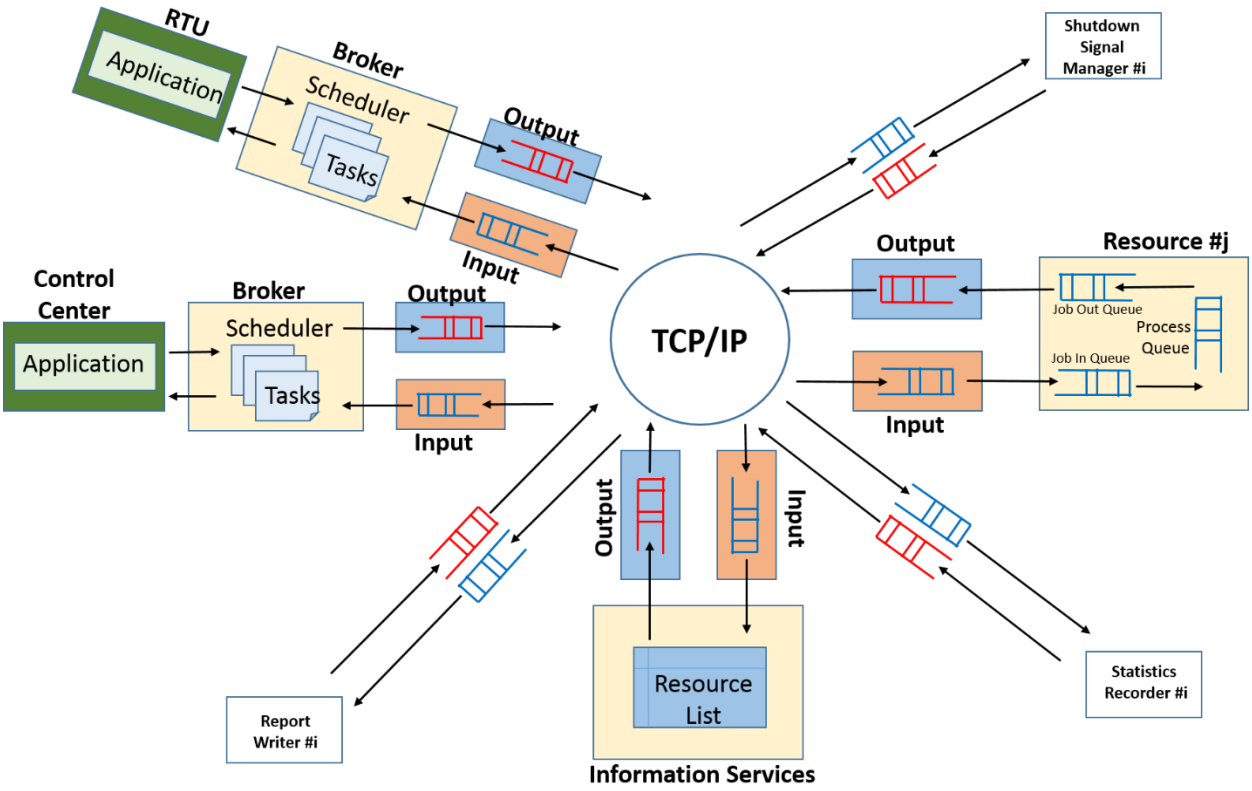

Figure 39 - A flow diagram in GridSim-based simulations

6.5. Control Center and RTU Application Model

No specific application model is defined in the GridSim toolkit; therefore, we developed an application model to meet the requirements of the CPSA co-simulator. The application model sends the execution signal to the CPSA module to initiate cyber-physical security evaluation, which involves performing observability analysis, state estimation, contingency analysis, and security metric computation functions. Each task instantiated in the Application Model has its characteristics and requirements defined through Gridlet objects. A Gridlet is a package that contains all the information associated with a task and its execution details, such as the job length expressed in MIPS, the disk I/O operations, the size of input and output files, and the task

originator. These parameters determine both the execution time and the time required to transport the input and output files between the control center and the RTUs.

6.6. Communication Protocol Model

Events trigger the interaction among GridSim entities. Events fall into two broad categories: service request and service delivery. Events can be implemented as either internal or external and as synchronous or asynchronous. Events originating from the same entity are called internal events, and those originating from the external entities are called external events. Synchronous events are ones when the event source entity waits until the event destination entity is finished performing all the tasks associated with the event. The control center polling the RTUs is implemented as a synchronous event. Asynchronous events are those in which the source entity initiates an event and continues with other activities without waiting for the completion of the initial event. In a typical CPSA co-simulation, the complete set of entities and the use of events for simulating the interaction between the entities are shown in Figure 40. Figure 40 depicts the interaction between a resource entity that simulates time-shared scheduling and other entities. The GridSim entities such as the control center and RTU users, resource broker, information service, statistics, shutdown, and report writer send events to other entities to signify the request for service, to deliver results, or to raise internal actions.

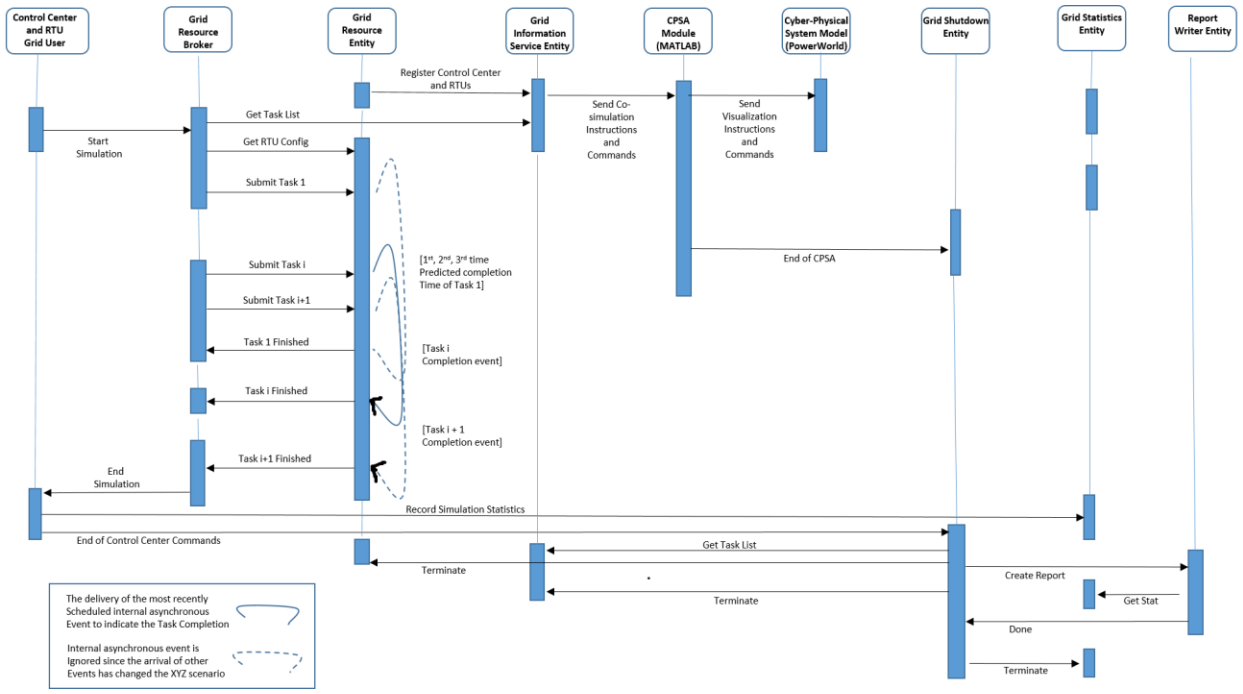

Figure 40 - An event diagram for the interaction between a time-shared resource and other entities

6.6.1. Scheduling of Time-Shared Resources

Time-sharing instead of space-sharing of resources, was implemented in the GridSim simulator. The resource simulator uses internal events to simulate the execution of Gridlet tasks, which are initiated by the control center when the control center polls the RTUs for data. At the beginning of the simulation, 24 tasks/jobs are transmitted to the RTUs for a 24-substation power system where each substation has a dedicated RTU. When jobs arrive, time-shared systems start their execution immediately and share resources among all the jobs. Time-sharing is executed through a round-robin algorithm. Whenever a new Gridlet task arrives, the processing time of the existing Gridlets is updated, and then the newly arrived job is added to the execution set. Figure 7 shows the algorithm for simulating the time-shared scheduling and execution. The algorithm is

adapted from [113] and extended in this research work:

Algorithm: Time-Shared Grid Resource Event Handler()

1. Wait for an event from control center or RTU
2. If event is external and RTU arrival task, then:
   BEGIN /*a new task arrived*/
   a. Allocate resource Share for RTUs Processed so far
   b. Add new RTU's task to Execution Set
   c. Forecast completion time of all tasks issued by RTU in Execution Set
   d. Schedule a task to be delivered at the smallest completion time
   END
3. If event is internal and its tag value is the same as the recently scheduled internal event tag,
   BEGIN /*a task finish event*/
   a. Allocate resource share for RTUs Processed so far
   b. Update finished RTU's resource parameters and send it back to the RTU's broker
   c. Remove finished RTU from the Execution Set and add to Finished Set
   d. Forecast completion time of all RTU in Execution Set
   e. Schedule a task to be delivered at the smallest completion time
   END
4. Repeat the above steps until the end of simulation event is received

Figure 41 - An event handler algorithm for scheduling time-shared resources

6.7. GridSim Java Package Design

A unified modeling language (UML) diagram of the modified GridSim package class diagram hierarchy is presented in Figure 42. The specification of each class contains up to three parts: attributes, methods, abstract and internal classes. The GridSim package implements the following classes, which were adapted from [113] and extended in this research work:

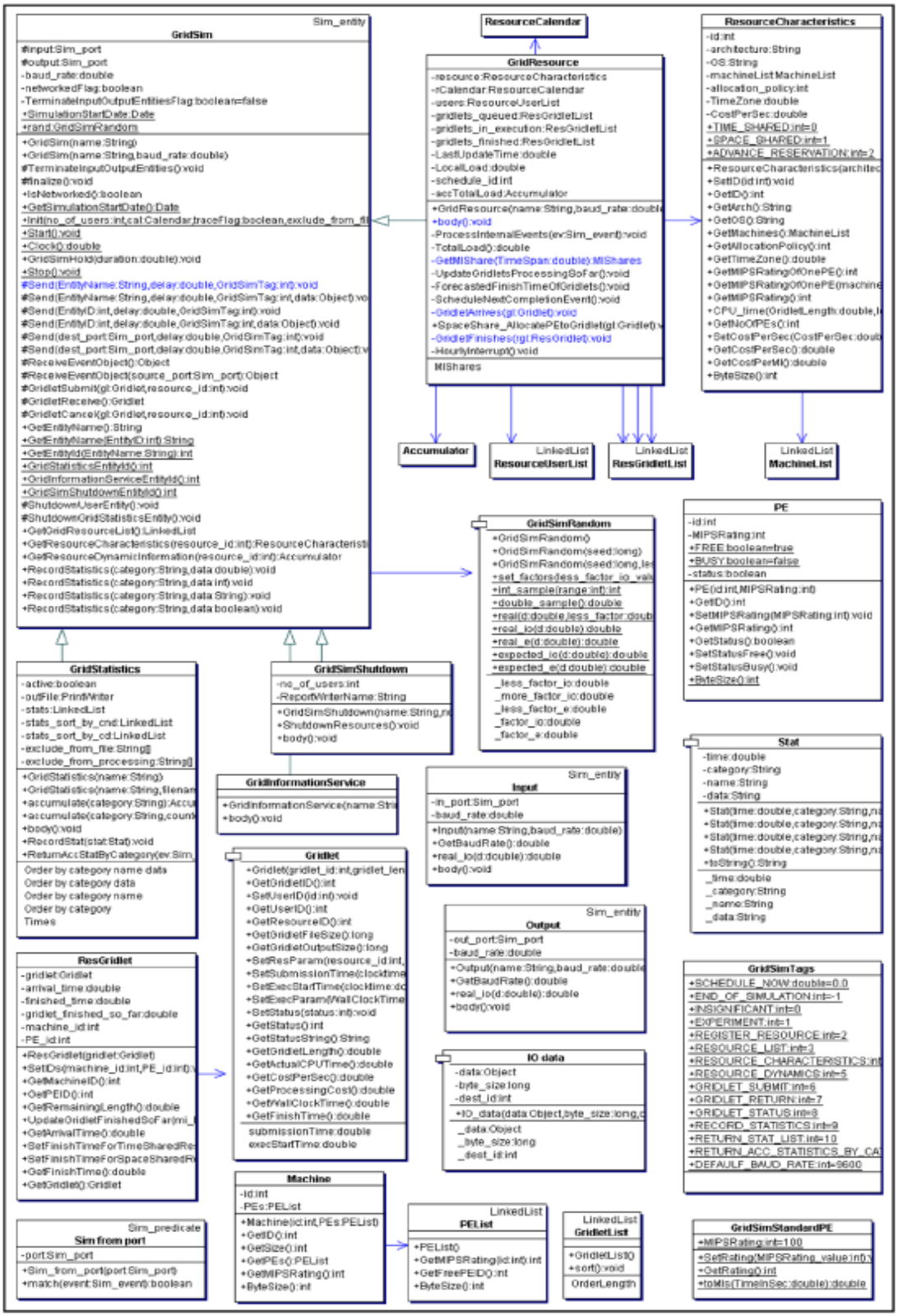

Figure 42 - Class Diagram of GridSim Package using UML Notation [113]

class gridsim.GridSim: This class is extended by all GridSim entities. It is the main class of the GridSim package and inherits event management and threaded entity features from the **eduni.simjava.Sim** entity class. The GridSim class provides networking and event delivery features, which allow synchronous communication for service access or delivery. A networked GridSim entity gains communication capability via the objects of GridSim's I/O entity classes, **gridsim.Input** and **gridsim.Output** classes. The GridSim class supports methods for simulation initialization, management, and flow control.

class gridsim.Input: This class was implemented to extend the **eduni.simjava.Sim** entity class and instantiates a port through which a simulation entity receives data. It maintains an event queue to serialize the data-in-flow and delivers the data to its parent entity. Inputs include communication parameters such as the baud rate, propagation delay, number of packets, and packet size. Power system input parameters include Bus, Branch, Generator, Load, and Shunt parameters. Input parameters are first sanitized to remove any errors during the input process and then held by an array list data structure from which they are sent to the power system and communication network configurators that then use the information to configure and set up the different systems to be simulated.

class gridsim.Output: Similar to the **gridsim.Input** class, the **class gridsim.Output** instantiates a port through which a simulation entity sends data to the simulated network. It maintains an event queue to serialize the data-out-flow and delivers the data to the destination entity.

class gridsim.Machine: This class defines a uniprocessor or shared memory multiprocessor machine. In this class, all RTUs and the control center are modeled as machines.

class gridsim.MachineList: An instance of this class simulates a collection of RTU machines. A star topology is used to connect all the RTUs to the control center in a Wide Area Network (WAN) architecture.

class gridsim.GridResource: This class extends the GridSim class and inherits the communication and concurrent entity capability of the GridSim class. An instance of this class simulates a resource with properties defined in an object of the **gridsim.ResourceCharacteristics** class.

class gridsim.Gridlet: This class represents a job package that contains job length, the length of input and output data in bytes, execution start and end time, and the originator of the job.

class gridsim.GridletList: This class is used to maintain a list of Gridlets and support methods for organizing them.

6.8. Designing CPSA Interface(s) Connections

This section describes the design of special interfaces and Application Programming Interfaces (APIs) that are used in the software development of the co-simulator.

6.9. Designing the MATLAB-PowerWorld Connection Interface:

This interface is established with the use of the COM server offered by *SimAuto*. Through this interface, PowerWorld can be requested to run instructions such as the following:

1. Open, save, and close a case (network).
2. List the devices of each type (buses, branches, generators, loads, etc.) present in the case.
3. Get the parameters (status, MW and MVAR rating, nominal voltage, etc.) of different elements or all elements of a given type.
4. Change the parameters of an element or all elements of a given type.
5. Run a power flow using the Newtown-Raphson method.

Once a connection between MATLAB and PowerWorld is established, functions running the above-mentioned instructions can be used in MATLAB to interact with PowerWorld and return results.

6.9.1. Designing the JADE-PowerWorld Interface

JADE cannot directly interface with PowerWorld. It must connect through MATLAB. There is no Java documentation available to directly connect Java with PowerWorld as a COM object. To enable a connection between JADE and MATLAB, we use a Transmission Protocol Connection (TPC), which allows us to run both MATLAB and PowerWorld on a single computer or using a remote computer for running MATLAB and PowerWorld. This connection procedure was adapted from [114] and extended in this research work. The connection between MATLAB and PowerWorld is established with a COM object through *SimAuto*. A single agent in JADE handles all communications with MATLAB using *InterfaceAgent*. On initialization, a TCP connection is established between *InterfaceAgent* and MATLAB and is then open throughout the

entire simulation. The JADE agent sends a message with the desired action information to *InterfaceAgent* using the standard Message Transport Protocol (MTP). *InterfaceAgent* processes the content of the message and sends it to MATLAB through TCP. MATLAB receives the message, processes the respective parameters, and requests PowerWorld to run the appropriate instructions. After executing the instructions, PowerWorld returns the result to MATLAB through the COM interface. MATLAB then reprocesses the answer and sends it through TCP back to *InterfaceAgent*. Finally, *InterfaceAgent* processes the result it received and sends the final response to the agent that issued the initial request.

The run times indicated in Figure 43 show that these interfaces perform well to get (obtain) the parameters of the buses and run a power flow on a 42-bus test system. Note that run times may vary for other systems and other runs.

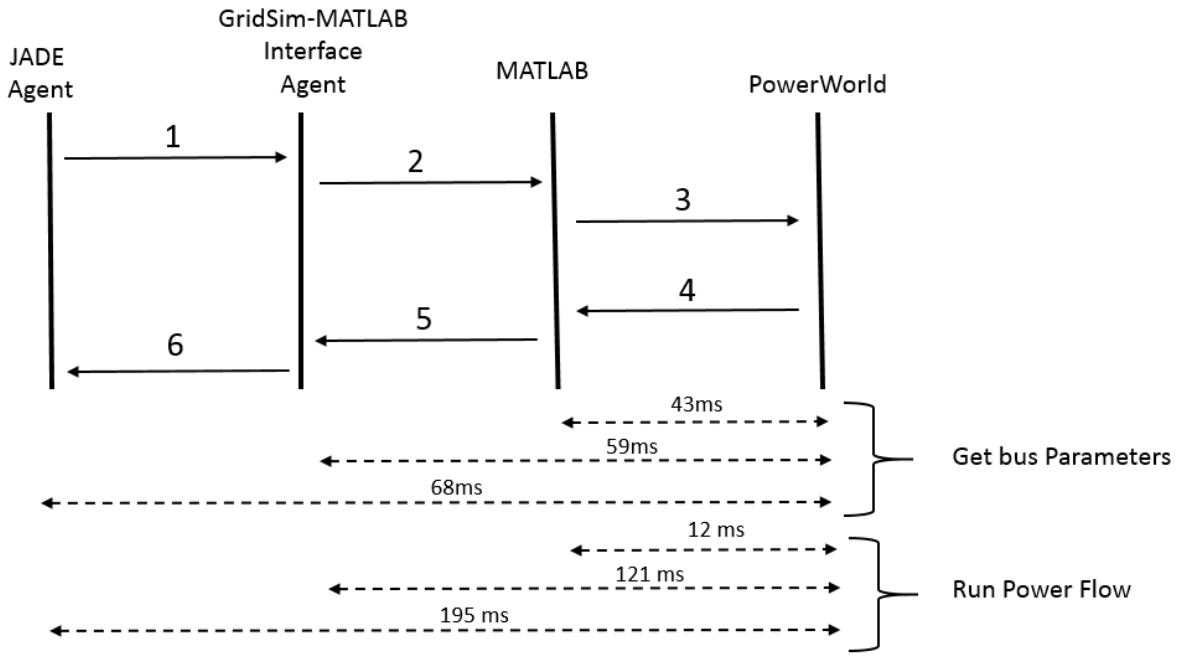

Figure 43 - Communication flowchart of a request issued by a JADE agent

6.10. Contribution and Conclusion

This chapter presented the software implementation framework of the co-simulator tool developed for assessing bulk electric system cyber-physical security. One contribution of Chapter Six is an explanation and presentation of:

1. The design and implementation of a novel GridSim-MATLAB interface, which acts a gateway for message passing communications between the two platforms.
2. The design and implementation of a novel MATLAB-PowerWorld interface for executing real-time scripting instructions in PowerWorld through the PowerWorld automation server.
3. The design and implementation of a modular architecture for simulating the communication network of the test cyber-physical systems by using GridSim.
4. The implementation of a cyber-physical security assessment module which efficiently and robustly characterizes and quantifies the cyber-physical security of the system under normal and attack conditions.

CHAPTER VII

CYBER-PHYSICAL SECURITY ASSESSMENT

The cyber-physical security assessment (CPSA) co-simulator tool developed in this dissertation accesses the cyber-physical security of an entire cyber-physical bulk electric system. Also, it allows management of the communication network (by controlling the baud rate, propagation delay, and Maximum Transmission Unit (MTU)) and the substations topology (the connection of nodes and routers).

7.1. Bad Command Injection Attack Impact Evaluation

One way that we can observe the cyber-physical impact of a bad command injection attack on the power system is by using different setting preferences in the Wide Area Communication Network (WACN). The adversary can either send a malicious command to the substation's RTU or alter a legitimate command that was sent by the control center over an insecure network. To test the co-simulator tool presented in Chapter Six, we used it to perform a cyber-physical security assessment on two different 24-substation power system with 42 buses, 62 lines, 8 generators, 27 loads, 6 transformers, and 9 shunt capacitor banks. The tool first simulates the normal operation case before simulating the bad command injection attack scenario. When simulating normal operation, CPSA is initialized with the main interface, as shown in Figure 45, where the operator sets communication parameters such as the baud rate, propagation delay, number of packets to be sent, and packet size window at the Control Center (CC)and the RTU. We simulated a normal case to provide the operator with a baseline case with which to compare the bad command attack scenario against. Deviations from normal operational trends help the operator identify when the system is under attack. The main interface provides the operator with the evolution of the insecurity

of the system over time. The polling requests (a read command) from the CC to different substations' RTUs are initiated every 5 seconds. Upon receiving a request, each RTU acknowledges the request and starts the process of collecting field measurements. Then each RTU prepares to send the measurement value packets over the wide-area network. Similarly, once the CC receives these packets from the RTU, the CC sends an acknowledgment to each respective RTU. The sent power system measurements include active and reactive line power (LineMW, LineMVR), bus voltage and angle (BusPUVolt, BusRad), generator active and reactive power (GenMW, GenMVR) and voltage (GenVolt), load active and reactive power (LoadMW, LoadMVR), and transformer tap ratio (LineTap). After receiving the measurements, the operator performs control actions to balance the demand-supply of power. These actions include changing the status (open/close) of circuit breakers connected to various power system components, such as transmission lines (LineStatus), generators (GenStatus), loads (LoadStatus), transformers (modeled as LineStatus), and shunt capacitors (SSStatus)

7.1.1. Test Case A

Test Case A simulates a bad command injection attack on the cyber-physical bulk electric system as shown in Figure 44.

Pre-conditions: Using load forecast information derived from historical data, we simulate the expected normal operational behavior of the power system (under no attack) for eight iterations, as shown in Figure 45. The result of this simulation is stored in comma-separated files and is adopted as the baseline case. The simulation lasts for eight minutes with 8 time steps, each of

which represents a one-minute interval. Next, the bad command injection attack is simulated. An attack can target at any time step.

Main Process: We use the co-simulator to assess the effect of a bad control command over the communication network. Below are the execution steps of the co-simulation:

1) The co-simulator models a communication scenario, which maps the real communication network parameters. These parameters are set as follows: the baud rate = 1572864 bits/sec, the propagation delay = 300 ms, command = "CC → RTU: Send Measurement Values-," the packet buffer size at the CC = 180 bytes, and the packet buffer size at the RTU = 1500 bytes.

2) We model and perform a single attack as a malicious command injection operation, and the command type "Change Line Status" is set from "close" to "open," and the command is transmitted to the RTU that changes the status of the transmission line from bus 32 to bus 37 from "closed" to "open" via the attack modeler in Figure 46

3) The IDS suspects the command is not a "legitimate command" based on its rules filtration and pattern matching. The IDS then notifies the operator about a malicious command based on its rule formation and command pattern matching. Figure 47 shows an "ALERT" dialog box that indicates that the IDS suspects the command to be malicious. The operator can view the JSON (JavaScript Object Notation) file and its parameters. The role of the IDS is critical here because it notifies the operator about a cyber incident, which is not a normal power routine operation.

4) The operator at the CC views the command log information and finds that the command was not initiated by himself/herself. The operator then decides to simulate the command and observe its effect on the power system with the option to allow the command to be executed in the actual system or not, as shown in Figure 48.

Result: For normal operation, the aggregate megawatt contingency overload (AMWCO) value, which measures the system insecurity, is observed to be 200 MWCO, which is normal and acts as the baseline case. When the simulation is started, a malicious command is suspected by the IDS at time step 5. This is because, under the bad command injection case, the adversary injected a malicious command into the system at timestep 5. The IDS sends an alert message through the IDS notification interface. This alert message prompts the operator for the next action to either simulate or reject the command, as shown in Figure 47 and Figure 48. If the operator decides to simulate the effect of the malicious command, this generates an attack output file and finally allows the execution of the command on the real power system with the effect observed in Figure 49 and Figure 50.

Post-conditions: Once a malicious command is successfully injected and simulated, the power system cyber-physical security is evaluated. Then, the results of the simulation of the bad command injection attack scenario are compared with the baseline case to characterize the system's deviations from the normal operations. An attack is identified if the real results differ significantly from the projected results. Figure 45 shows the evolution of the cyber-physical security of the system under the normal case plotted from time step 1 to 8 versus the total system load and the AMWCO. Figure 49 shows the system AMWCO when the bad command "Open line 32-37" is simulated in the power system at time step 5. At that iteration, the AMWCO suddenly increases and remains high for the remainder of the simulation. The effect of the attack can be observed at the attack trigger time of 5, as shown in Figure 49, which shows that the security of the system as measured in this test by the AMWCO increase by 50% from 200 MWCO under the

normal operation case to 300 MWCO as a result of the bad command injection that caused overloading of transmission lines around the 32-37 line connection. Figure 50 shows the visualization of the system under attack with the transmission line MW meter reading in red, indicating that the transmission line was overloaded. In this malicious command scenario, the components in the system under attack are highlighted for the operator through the use of a blinking alarm that is triggered when said components are maliciously tampered with. In this case, lines 32-37 blink. The communication lines used to access the breakers tied to lines 32-37 also trigger an alarm and blink. This feature of our visualization module allows the operator to react quickly to the portion of the system under attack and take remedial actions to mitigate the effects of the attack, in this case by reclosing lines 32-37.

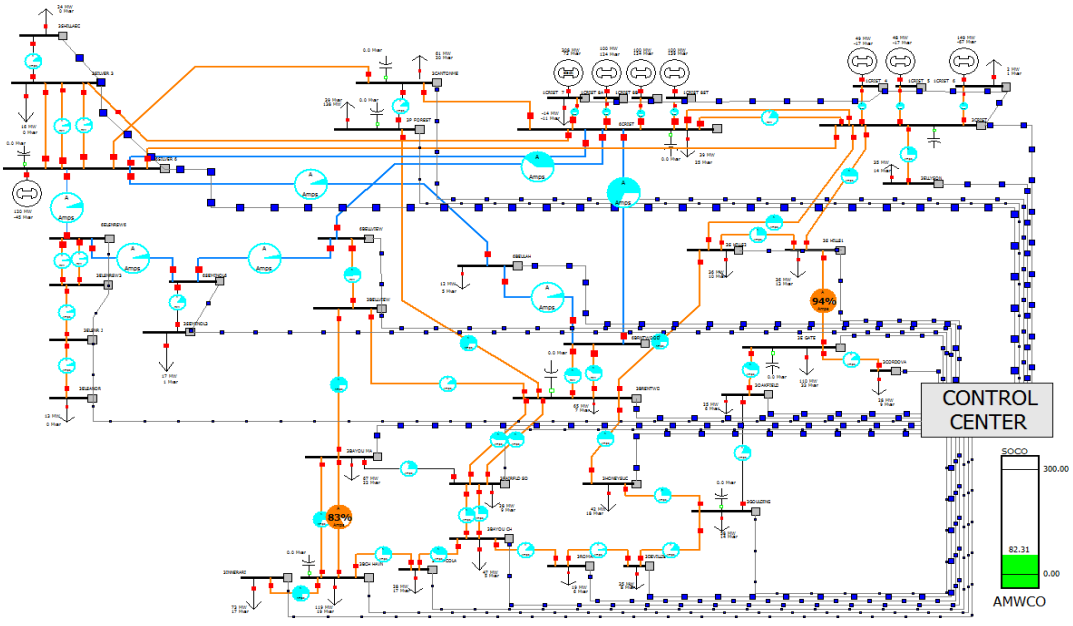

Figure 44 - 24-Substation Cyber-Physical System under Normal Operation

Figure 45 - CPSA main interface showing normal operation

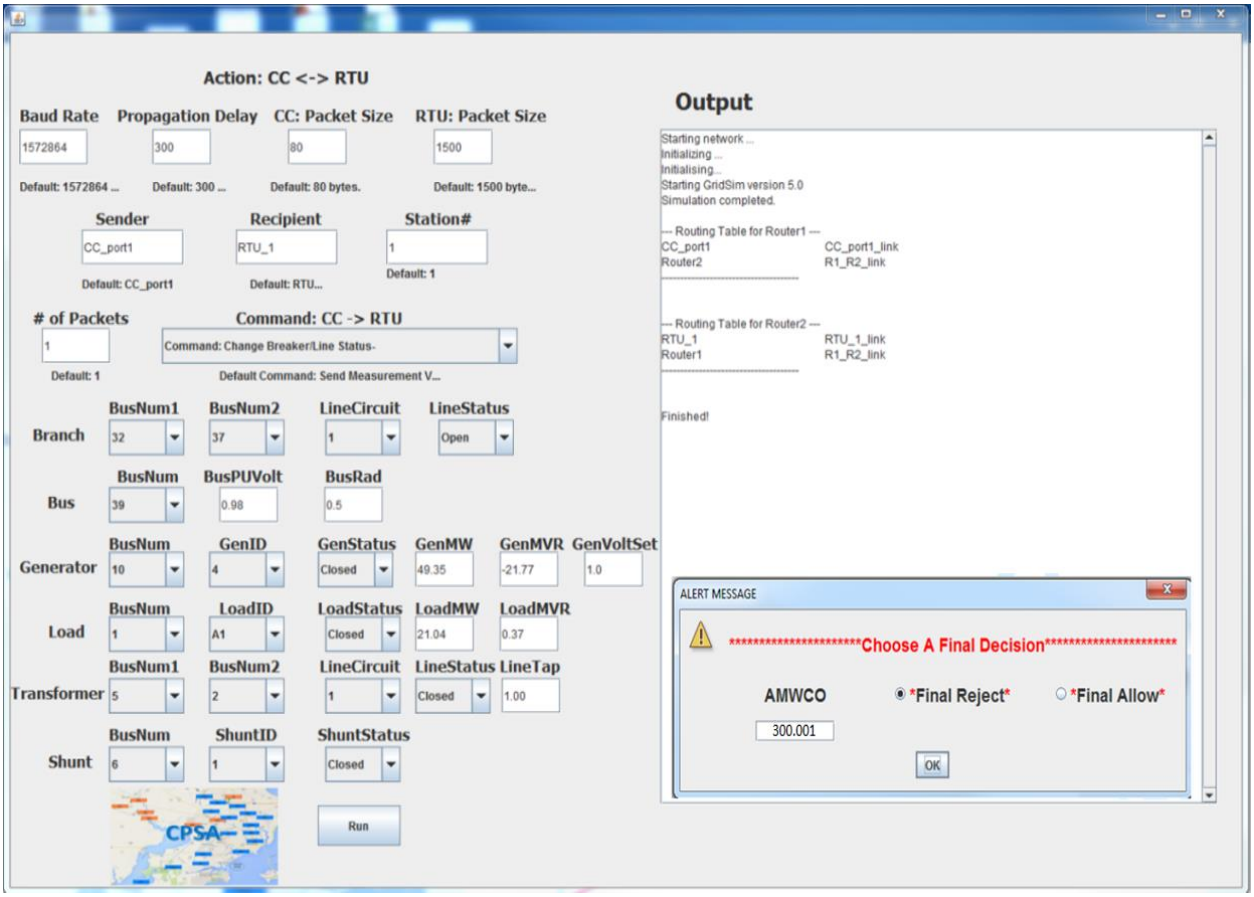

Figure 46 - Attack Modeler used for simulating a bad command injection attack.

Figure 47 - IDS Alert of a Bad Command with the Option to Simulate

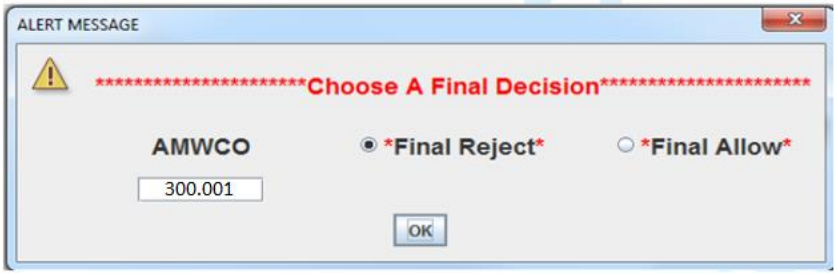

Figure 48 - Final decision to accept or reject the command

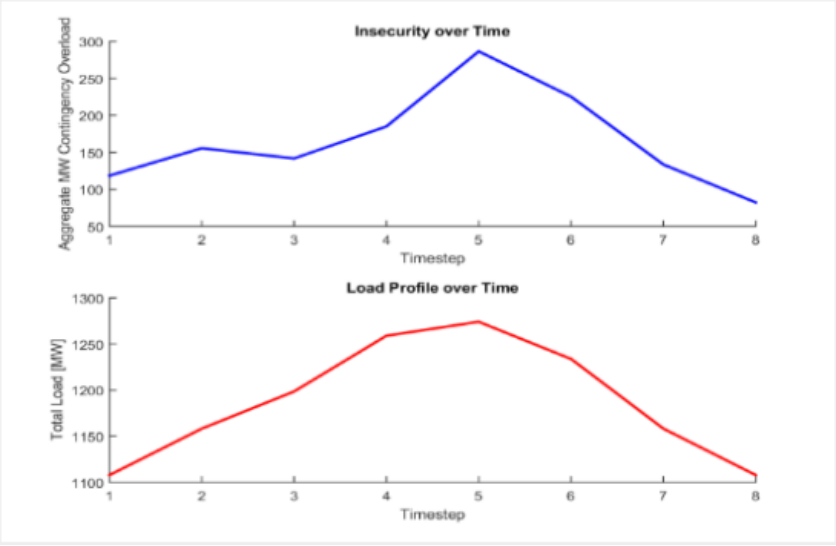

Figure 49 - Bad command injection (open line 32-37) on 24-Substation Test Case A System

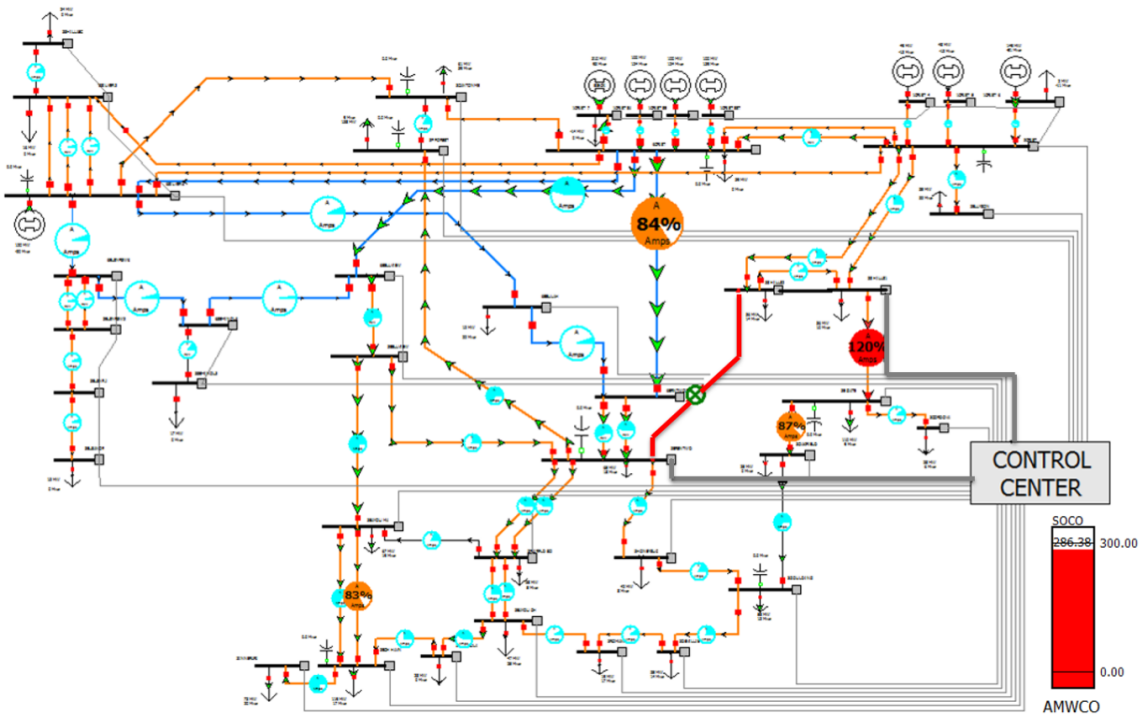

Figure 50 - CPSA visualization capturing lines 32-37 under attack and the communication channels used to access the breakers connected to transmission lines 32-37.

7.1.2. Test Case B

This section describes designing, implementing, and co-simulating a bad command injection attack in CPSA on a different 24-substation cyber-physical bulk electric system. The process for simulating the cyber-physical security assessment for this case is the same as that for test case A, but it results in different system post-conditions.

Post-conditions: A malicious command is successfully injected and simulated, and the system's cyber-physical security is evaluated. The results of the simulations of the bad command injection attack scenarios are compared with the results (those) of the baseline case to

characterize/determine the extent to which (how much) the system deviated from normal operations. An attack is identified if the real results differ significantly from the projected results. Figure 53shows the evolution of the cyber-physical security of the system under the normal case plotted against the total system load and also plotted against the AMWCO. Figure 56 shows the system AMWCO when the bad command "Open lines 32-37" is simulated in the power system at time step 5. The AMWCO suddenly increases at that iteration and remains high for the remainder of the simulation. This sudden increase demonstrates that the effect of the attack can be observed at the attack trigger time of 5. The security of the system as measured by the AMWCO increased by 37.5% from 8,000 MWCO under the normal operation case to 11,000 MWCO as a result of the bad command injection, which caused an overloading of transmission lines around the line 32-37 connection. Figure 52 shows the visualization of the system under attack with transmission line MW meter reading in red, indicating that the transmission line was overloaded. The components in the system under attack are highlighted for the operator through a blinking alarm that is triggered when said components are maliciously tampered with. In this case, line 32-37 blinked. The communication lines used to access the breakers tied to line 32-37 also triggered an alarm and blinked. This feature of our visualization module allows the operator to react quickly to the portion of the system under attack and take remedial actions to mitigate the effects of the attack, in this case, reclosing lines 32-37.

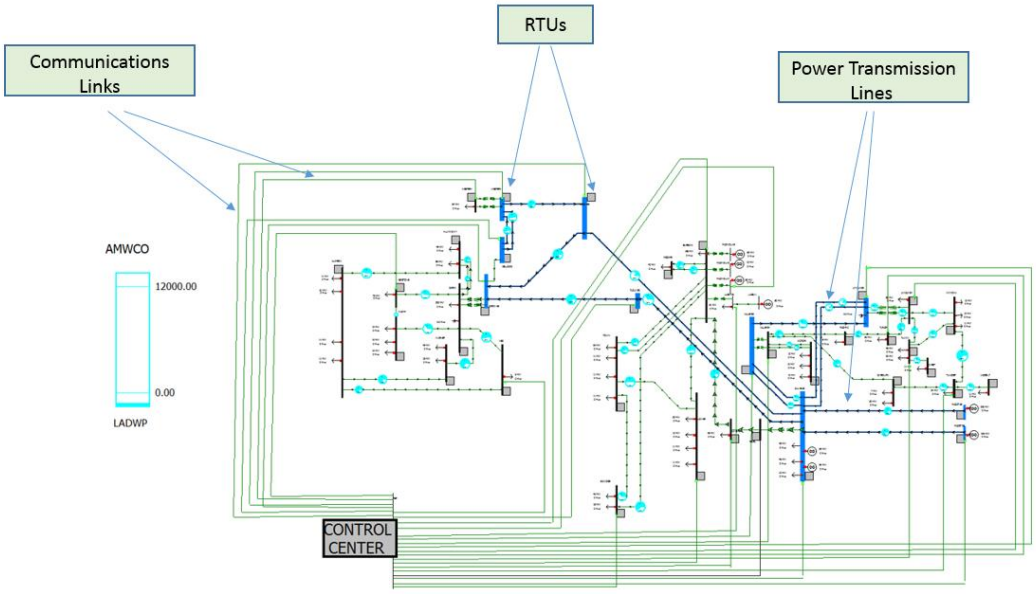

Figure 51 - Normal Operation of Test Case B System

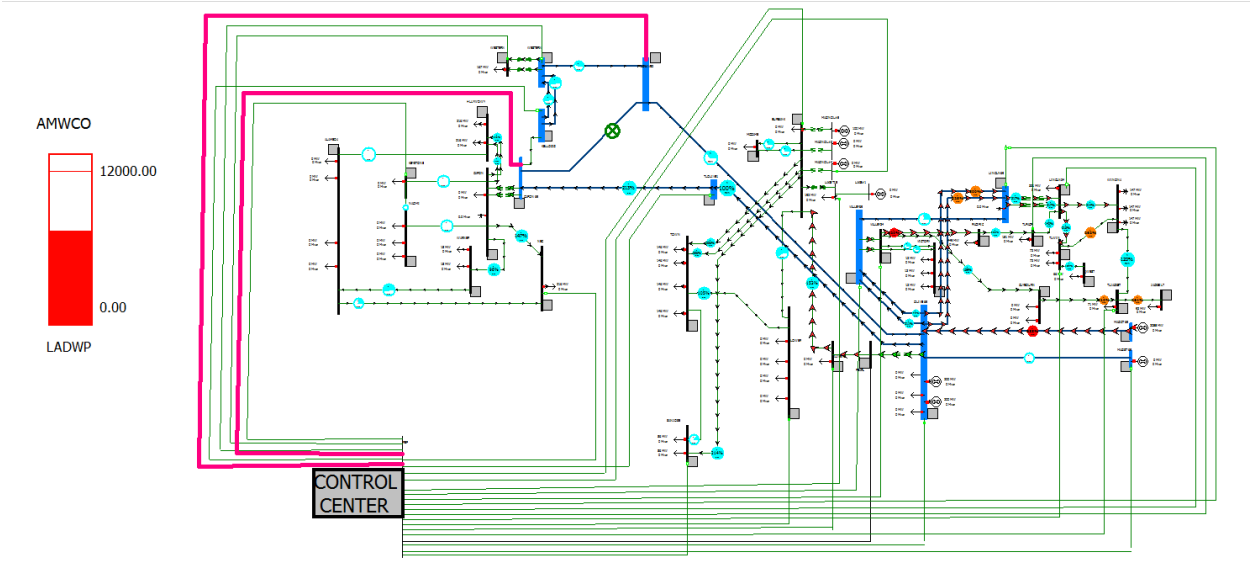

Figure 52 - Test Case B System under a Bad Command Injection Attack

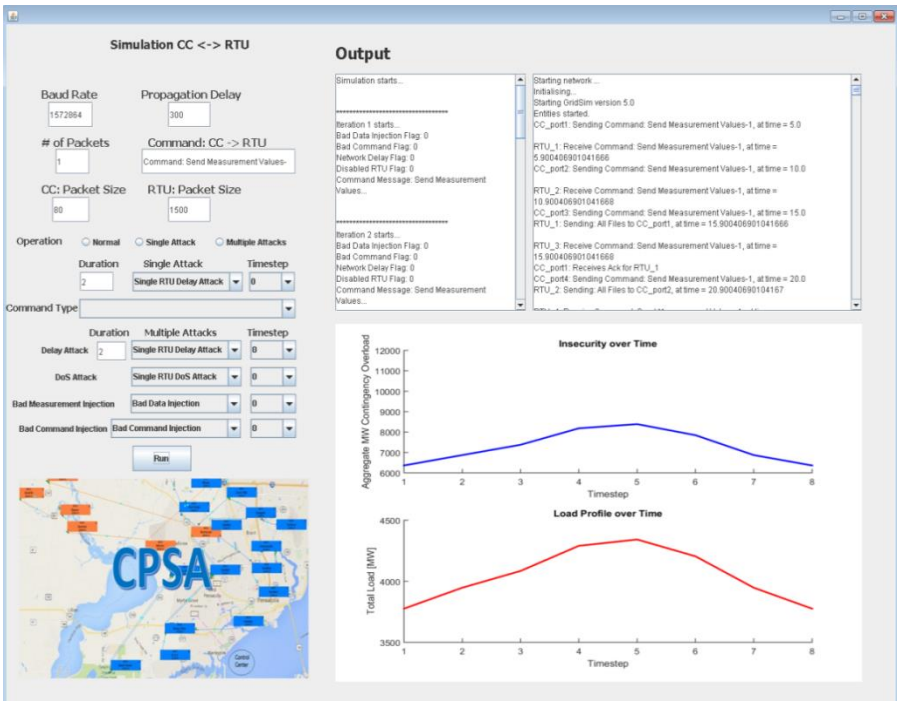

Figure 53 - CPSA main interface under normal operation for Test Case B System

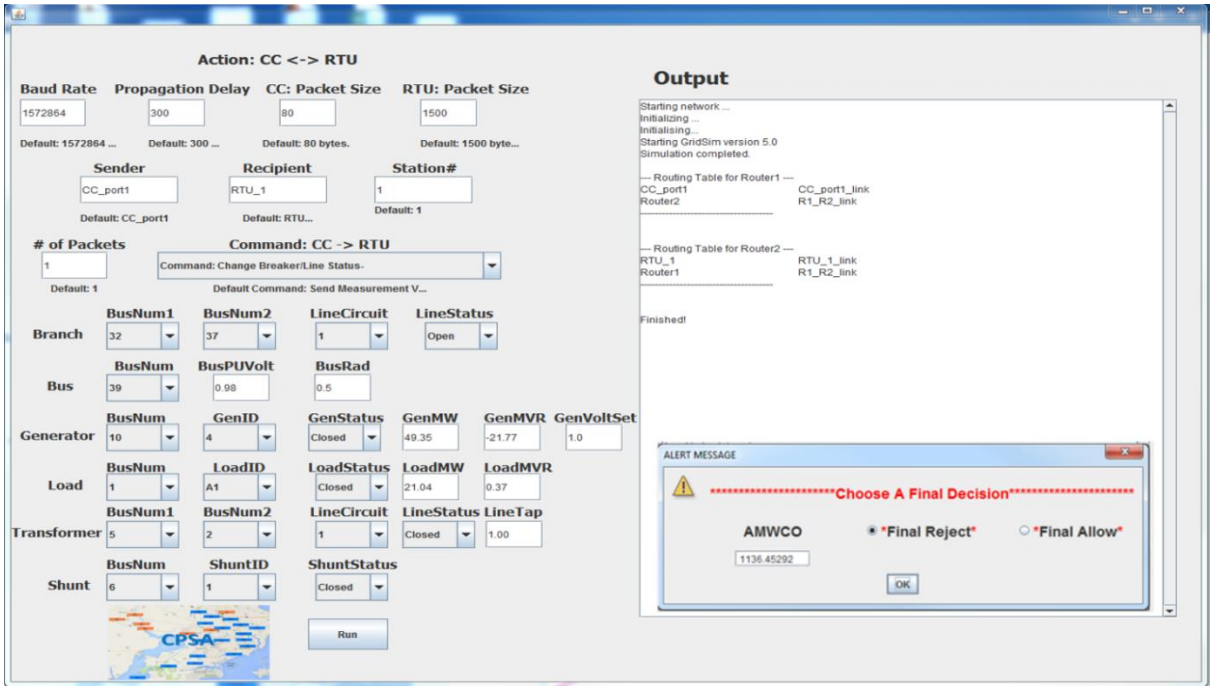

Figure 54 - Attack Modeler used for simulating a bad command injection attack.

Figure 55 - Simulation of a bad command by the operator at the control center.

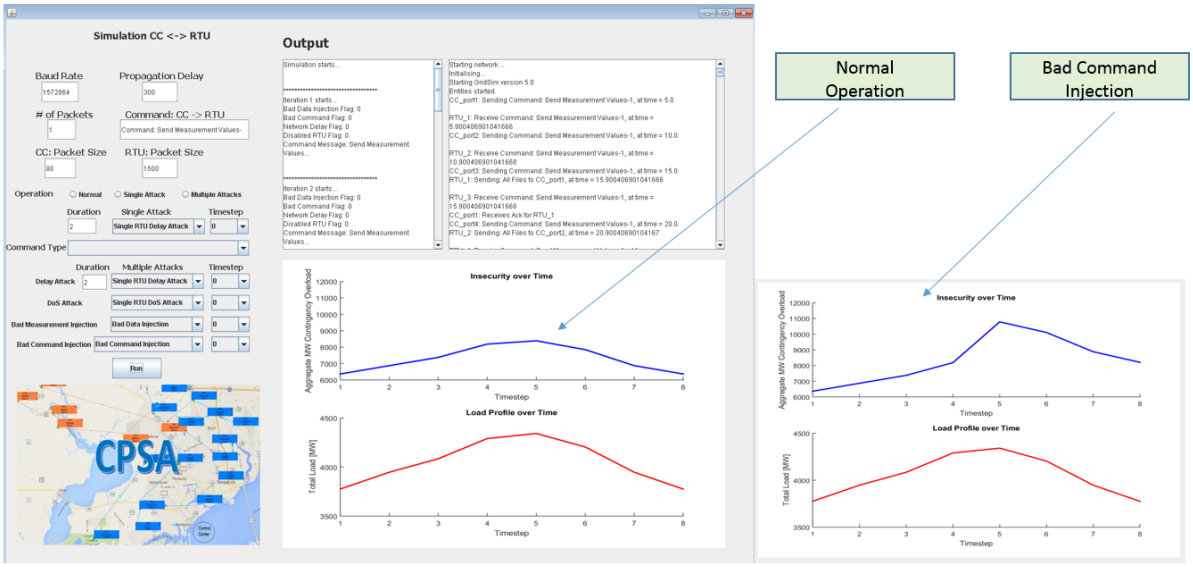

Figure 56 - CPSA visualization capturing lines 32-37 under attack and the communication channels used to access the breakers connected to transmission lines 32-37.

The plot shows the comparison between the baseline case on the left and the bad command case on the right.

7.2. Cyber Threat Capability Analysis

The threat capability metric keeps the historical details of suspicious threats, such as a malicious source or destination IP, and altered data or commands, as shown in

Table 11 below.

Table 11 - Threat Capability Matrix

Threat Suspect	Source IP	Destination IP	Timestamp	Data Type	Packet Size (Octets)
CC Port-10	192.168.0. 3	192.168.0.7	23-Oct-16, 10:15:27	substation data	255
RTU-6	192.168.0.7	192.168.0.13	31-Oct-16, 21:32:11	command "open Gen 6"	125
RTU-16	192.168.0.7	192.168.0.23	5-Nov-16, 11:45:37	command "open Load 21"	127
RTU-24	192.168.0.7	192.168.0.31	10-Nov-16, 18:10:23	command "open Trans 6"	122

The co-simulator also maintains event logs of the activities performed at the intermediate routers, substation RTUs, and control center. A sample event log at an intermediate router is shown in Figure 5.

2.0 receive router ad from	Router2										
5.3											
5.3 receive incoming	Packet#1	out of	1	with id	997260727	from	Output_CC_port1	to	RTU_1	tag	GridSimTags.delay 0
5.3 enqueing	Packet#1	out of	1	with id	997260727	from	Output_CC_port1	to	RTU_1	tag	GridSimTags.FLOW_SUBMIT
5.3 dequeuing	Packet#1	out of	1	with id	997260727	from	Output_CC_port1	to	RTU_1	tag	GridSimTags.FLOW_SUBMIT
10.3											
10.3 receive incoming	Packet#1	out of	1	with id	1721393242	from	Output_CC_port2	to	RTU_2	tag	GridSimTags.delay 0
10..3 enqueing	Packet#1	out of	1	with id	1721393242	from	Output_CC_port2	to	RTU_2	tag	GridSimTags.FLOW_SUBMIT
10..3 dequeuing	Packet#1	out of	1	with id	1721393242	from	Output_CC_port2	to	RTU_2	tag	GridSimTags.FLOW_SUBMIT
15.3											
15.3 receive incoming	Packet#1	out of	1	with id	339570773	from	Output_CC_port3	to	RTU_3	tag	GridSimTags.delay 0
15.3 enqueing	Packet#1	out of	1	with id	339570773	from	Output_CC_port3	to	RTU_3	tag	GridSimTags.FLOW_SUBMIT
15.3 dequeuing	Packet#1	out of	1	with id	339570773	from	Output_CC_port3	to	RTU_3	tag	GridSimTags.FLOW_SUBMIT

Figure 5: Event logs maintained at the intermediate routers

Efficient and accurate maintenance of logs equips the operator with adequate historical data to perform a compare and contrast analysis of current events in real-time with past events and helps the operator make informed attack mitigation decisions.

7.3. Performance Analysis

We developed an experimental setup with a CPSA co-simulator that was implemented through multiple platforms, namely GridSim, JADE, MATLAB, and PowerWorld, to simulate power monitoring and control and cyber-physical attack scenarios between the control center and the substation RTUs. The C37.118 protocol format was adopted to implement data communication through message passing in the co-simulator.

7.4. Overhead

The overhead generated by the proposed approach includes packet scans and deep packet inspection by the IDS and command simulation by the operator if the IDS flags the command as malicious.

7.5. Scalability

The proposed approach can detect single as well as multiple malicious commands targeting power system components. In this dissertation research, we simulated and tested our approach on 24-substation systems, with each substation having its own dedicated RTU. However, the co-simulator can be used to test much larger systems with more substations and hundreds of buses.

7.6. Robustness

In general, the robustness of the co-simulator depends on the ability of the IDS to flag malicious commands as suspicious. Even if the IDS does not detect a malicious command and the command is executed on the real system, our approach can detect the power system disturbance

and report the effect to the operator, who can take appropriate actions (such as sending other control commands) to diminish the impact of the malicious command.

7.7. Execution and Response Time

The co-simulator simulates normal operations for 8 time steps by using the current-day next 8 minutes load forecast. Each time step represents a one-minute interval, and the generated output for each iteration is stored in a file. The co-simulator runs faster than real-time in the sense that during real-time operation, the system compares the actual output against the simulated output parameters (AMWCO and other system metrics of measurements). Our co-simulator generates each output file in less than three seconds, which is important in ensuring a fast response time of the power system operator.

7.8. Limitations

The limitation of the proposed approach is that it has been tested only by using single and sequential malicious attacks. The proposed approach does not currently support the detection of coordinated attacks.

7.9. Contributions and Conclusion

The contributions of the cyber-physical security assessment tool are highlighted below.

1. The developed tool helps power system operators better learn about and understand the nature of cyber-physical attacks, specifically bad command injection attacks targeting critical bulk electric system components.

2. The tool provides a broad understanding of different network topologies to improve our analysis of communication network parameter settings for the bulk electric power systems.

3. The tool supports our understanding and evaluation of power system functions and routine operations, such as power flow and contingency analysis.

5. The tool helps us/analysts monitor and evaluate the impact of cyber-physical attacks on the physical power system. This provides a greater understanding of the impact on individual power components as well as on the power system as a whole. The tool maintains communication logs, received and sent power system measurement data, and triggered control commands. It also generates security metrics that pinpoint the critical and non-critical components in the system.

6. The tool will help operators make appropriate control decisions by simulating the impact of potentially malicious commands on the power system in real-time. It will enable operators to further develop their decision-making skills.

In conclusion, the work presented in this chapter describes an approach that uses a novel co-simulator to help us understand the potential impact of malicious command-based cyber-attacks on a power system. The generated output files, metrics, and graphs help the operator to understand changes in power system behavior in the presence of cyber-attacks. The detection of a malicious command takes place in real-time, which allows the operator to quickly respond to protect the system from malicious events in order to prevent cascading failures and eventually blackouts. In the future, the co-simulator can be used to test significantly larger power systems, those with a large number of communication network nodes. Further work in this area will extend the proposed approach to improve our understanding of the impact of coordinated attacks on the power system.

CHAPTER VIII

CONCLUSION

8.1. Discussion of Contributions

Currently, an operator at the control center can monitor power system line outages and other power system events. However, the operator has no knowledge of the cyber-physical security of the system. An adversary can perform cyber-attacks over the communication network to alter the transmitted measurement data or the critical commands, and in most cases, the operator will be unable to detect the attacks. Therefore, we need smarter tools and techniques to detect cyber-physical attacks over the bulk electric system. In this dissertation, we (1) developed algorithms that capture how bad data injection attacks propagate in a power delivery system, (2) developed a tool that models a bad command injection in bulk electric systems, and (3) developed a cyber-physical metric for quantifying the effect of a cyber-physical attack on bulk electric systems.

The first segment, Chapters 1-4 of this dissertation, presents the introduction, literature review, and the development of a graph-based attack propagation model that simulates a bad data injection attack and executes a heuristic defense strategy by using power system state estimation.. The state estimator was used to identify maliciously injected data and adopt physical security metrics to select appropriate attack mitigation actions. Visualization from the analysis performed by the propagation simulation can guide the operator at the control center to take appropriate action to minimize disruption of the physical power system operation due to the bad data injection attack.

The second segment, Chapters 5-7, explained the development, prototyping, testing, and evaluating of a co-simulator tool capable of modeling and simulating the effects of a cyber-physical attack such as a bad command injection attack and proposed recommendations for countermeasures against such attacks. The co-simulation involved a combined modeling of the

communications network (cyber layer) and the bulk electric system (power layer). Chapter 5proposed a system architecture covering the functional requirements and system modules of the developed co-simulator and described the dependencies and implementation of the co-simulator by using Java, MATLAB, and PowerWorld. The developed co-simulator supports the transmission of measurement data through polling request and response, triggering a control command to a power component deployed at a substation, and updating the power system values: voltage, active power, reactive power, and angle. We adopted the Aggregated Megawatt Contingency Overload (AMWCO) as the cyber-physical security metrics to quantify and qualify the real-time performance of the bulk power transmission system during a bad command injection attack. The second segment of the dissertation research makes three main contributions:

1. It models a bad command injection cyber-physical attack on the operation of a 24-bus bulk power transmission system through co-simulation.
2. It quantifies the cyber-physical security of the test system under a cyber-physical attack in real-time.
3. It increases operator situational awareness of system-level cyber-physical security.

The first contribution above explores modeling a bad command injection attack and studying the effect(s) of the attack on power system functions such as state estimation and contingency analysis. The second contribution explores using a cyber-physical security metric known as the Aggregated MegaWatt Contingency Overload (AMWCO) to capture the effect of overloading transmission lines as a result of a bad command injection attack. The third contribution of the dissertation addresses the visualization of the effects of the cyber-physical security attack. A novel process user interface was designed to capture the spatiotemporal dynamics of the bulk power system under the bad command injection attack. Currently, electric utility operators at the control center

do not have the means of quantifying and visualizing the effects of an on-going cyber-physical security attack. To potentially solve this problem, this dissertation work created a software tool that is intended to help utilities not only gain a better understanding of how the system behaves during a bad command injection attack but also be able to take the appropriate countermeasures to mitigate the effects of the attack. This software tool will improve operator training regarding cybersecurity by strengthening their awareness and understanding of the power system's behavior in the presence of potential cyber-attacks. It will also enable the operator to further develop their decision-making skills.

8.2. Future Work

Insights gained from the work presented in this dissertation can be useful for exploring the following future research directions of coordinated cyber-physical attacks in bulk electric systems:

1. Modeling and simulating multiple bad data injection attacks from several substations.
2. Modeling and simulating multiple bad command injection attacks that simultaneously disconnect a line and a generator.
3. Modeling and simulating N-1 RTU Contingency Analysis. Traditionally, contingency analysis has involved N-1 contingency analysis for transmission lines. This work can be extended to consider N-1 contingency analysis for RTUs. RTUs disabled as a result of a cyber-physical security attack can create severe observability problems in bulk electric systems and lead to a blackout. Hence, insights gained from this dissertation can help create a formulation and solution methodology for mitigating the effects of a cyber-physical security attack that results from a malicious actor disabling several RTUs in the bulk electric system.

REFERENCES

[1] F. C. Schweppe and D. B. Rom, “Power System Static-State Estimation, Part II: Approximate Model,” *IEEE Trans. Power Appar. Syst.*, vol. PAS-89, no. 1, pp. 125–130, 1970.

[2] “Communications network solutions for smart grids Smart,” *Siemens AG Infrastruct. Cities Sect.*, 2011.

[3] M. C. Sorebo, Gilbert N. and Echols, *An End-To-End View of Security in the New Electrical Grid*, 1st ed. Boca Raton, 2011.

[4] G. Flisberg, “Global Trends in Bulk Power Transmission,” 1970.

[5] IEEE, *C37.1-1987 - IEEE Standard Definition , Specification , and Analysis of Systems Used for Supervisory Control , Data Acquisition , and Automatic Control*, vol. 1994. 1987.

[6] “SCADA Systems for Electrical Distribution,” *Electrical Technology*. [Online]. Available: https://www.electricaltechnology.org/2015/09/scada-systems-for-electrical-distribution.html.

[7] “SCADA RTU’s.” [Online]. Available: http://members.iinet.net.au/~ianw/rtu.html. [Accessed: 01-Jan-2017].

[8] R. Carlson, “Sandia SCADA Program High-Security SCADA LDRD Final Report,” *Prod.Sandia.Gov*, no. April, 2002.

[9] D. J. Gaushell and H. T. Darlington, “Supervisory Control and Data Acquisition.,” *Proc. IEEE*, vol. 75, no. 12, pp. 1645–1658, 1987.

[10] W. J. Ackerman and W. R. Block, “Understanding supervisory systems,” *Comput. Appl. Power, IEEE*, vol. 5, no. 4, pp. 37–40, 1992.

[11] J. Lloyd, “Security hardened remote terminal units for SCADA networks,” 2008.

[12] J. Jayasamraj, “SCADA Communication & Protocols,” pp. 1–8.

[13] D. Laird, “Substation SCADA- Small, Medium, or Large?” [Online]. Available: https://www.hallam-ics.com/blog/substation-scada-small-medium-or-large-.

[14] M. Berg and J. Stamp, “A Reference Model for Control and Automation Systems in Electric Power,” *Sandia Natl. Lab. Rep.*, no. SAND2005–1000C, pp. 1–7, 2005.

[15] V. H. Nguyen, T. Tran-Quoc, and Y. Besanger, “SCADA as a service approach for

interoperability of micro-grid platforms," *Sustain. Energy, Grids Networks*, vol. 8, pp. 26–36, 2016.

[16] E. Csanyi, "3 generations of SCADA system architectures you should know about." [Online]. Available: http://electrical-engineering-portal.com/three-generations-of-scada-system-architectures.

[17] T. Agarwal, "Know all about SCADA Systems Architecture and Types with Applications." [Online]. Available: http://www.edgefxkits.com/blog/scada-system-architecture-types-applications/.

[18] M. D. Hadley and K. A. Huston, "AGA-12 , Part 2 Performance Test Results," no. August, 2007.

[19] A. Abur and A. Gómez Expósito, *Power System State Estimation: Theory and Implementation*, vol. 24. CRC Press, 2004.

[20] K. P. Lien, C. W. Liu, C. S. Yu, and J. A. Jiang, "Transmission network fault location observability with minimal PMU placement," *IEEE Trans. Power Deliv.*, vol. 21, no. 3, pp. 1128–1136, 2006.

[21] U.S.-Canada Power System Outage Task Force, "Final Report on the August 14, 2003 Blackout in the United States and Canada: Causes and Recommendations," *System*, no. April, p. 238, 2004.

[22] A. Ashok and M. Govindarasu, "Cyber attacks on power system state estimation through topology errors," *IEEE Power Energy Soc. Gen. Meet.*, pp. 1–8, 2012.

[23] J. Jiang and Y. Qian, "Defense Mechanisms against Data Injection Attacks in Smart Grid Networks," *IEEE Commun. Mag.*, vol. 55, no. 10, pp. 76–82, 2017.

[24] R. Tsang, "Cyberthreats, vulnerabilities and attacks on SCADA networks," *Univ. California, Berkeley, Work. Pap. ...*, pp. 1–23, 2010.

[25] R. Langner, "Stuxnet: Dissecting a cyberwarfare weapon," *IEEE Secur. Priv.*, vol. 9, no. 3, pp. 49–51, 2011.

[26] R. M. Lee, M. J. Assante, and T. Conway, "Analysis of the cyber attack on the Ukrainian power grid," *SANS Ind. Control Syst.*, p. 23, 2016.

[27] *ICSA-11-231-01 - Inductive Automation Ignition Information Disclosure Vulnerability*, 4th ed. Butterworth-Heinemann.

[28] "NIST 800-82, "Guide to Supervisory Control and Data Acquisition (SCADA) and

Industrial Control System Security.” .

[29] E. Mills, “Details of the first-ever control system malware.” [Online]. Available: https://www.cnet.com/news/details-of-the-first-ever-control-system-malware-faq/.

[30] Siemens, “SIMATIC WinCC / SIMATIC PCS 7: Information about Malware / Viruses / Trojan horses,” 2011. [Online]. Available: https://support.industry.siemens.com/cs/document/43876783/simatic-wincc-simatic-pcs-7%3A-information-about-malware-viruses-trojan-horses?dti=0&lc=en-WW.

[31] R. McMilan, “Siemens: Stuxnet worm hit inddustrial systems,” 2010.

[32] P. Oman, E. Schweitzer, and J. Roberts, “Safeguarding IEDs, substations, and SCADA systems against electronic intrusions,” *2001 West. Power Deliv. Autom. Conf.*, no. April 2001, pp. 1–18, 2001.

[33] P. IEEE Power and Energy, “The Utility and Grid of the Future. What will it bring?,” vol. 14, no. 5, 2016.

[34] E. Smith *et al.*, “Going beyond cybersecurity compliance,” *IEEE Power Energy Mag.*, vol. 14, no. 5, pp. 48–56, 2016.

[35] R. Maynor and R. Graham, “SCADA Security and Terrorism: We are not Crying Wolf,” 2006. [Online]. Available: https://www.blackhat.com/presentations/bh-federal-06/BH-Fed-06-Maynor-Graham-up.pdf.

[36] R. Lemos, “SCADA system makers pushed toward security,” *2006*. [Online]. Available: https://www.securityfocus.com/news/11402.

[37] M. K. Reiter, “False Data Injection Attacks against State Estimation in,” pp. 21–32, 2009.

[38] S. Sridhar and G. Manimaran, “Data Integrity Attack and its Impacts on Voltage Control Loop in Power Grid,” pp. 0–5, 2011.

[39] O. Kosut, L. J. L. Jia, R. J. Thomas, and L. T. L. Tong, “Malicious Data Attacks on Smart Grid State Estimation: Attack Strategies and Countermeasures,” *Smart Grid Commun. (SmartGridComm), 2010 First IEEE Int. Conf.*, pp. 1–6, 2010.

[40] E. Bompard, C. Gao, R. Napoli, A. Russo, M. Masera, and A. Stefanini, “Risk Assessment of Malicious Attacks Against Power Systems,” vol. 39, no. 5, pp. 1074–1085, 2009.

[41] Y. Liu, P. Ning, and M. K. Reiter, “False data injection attacks against state estimation in electric power grids,” *Ccs*, vol. 14, no. 1, pp. 1–33, 2009.

[42] G. Hug and J. A. Giampapa, “Vulnerability assessment of AC state estimation with

respect to false data injection cyber-attacks," *IEEE Trans. Smart Grid*, vol. 3, no. 3, pp. 1362–1370, 2012.

[43] A. Rahman and H. Mohsenian-rad, "False Data Injection Attacks Against Nonlinear State Estimation in Smart Power Grids," no. 1, pp. 1–5.

[44] J. Kim, L. Tong, and R. J. Thomas, "Dynamic Attacks on Power Systems Economic Dispatch."

[45] M. A. Rahman and E. Al-shaer, "Impact Analysis of Topology Poisoning Attacks on Economic Operation of the Smart Power Grid."

[46] M. Farajollahi, S. H. Hosseini, and A. Safdarian, "Bad Data Injection as a Threat for Power System Security," no. Sgc, pp. 23–24, 2015.

[47] S. D. Antonio, L. Coppolino, and I. A. Elia, "Security Issues of a Phasor Data Concentrator for Smart Grid Infrastructure," pp. 3–8, 2005.

[48] L. Xie, Y. Mo, S. Member, and B. Sinopoli, "Integrity Data Attacks in Power Market Operations," vol. 2, no. 4, pp. 659–666, 2011.

[49] A. Giani, E. Bitar, M. Garcia, and M. Mcqueen, "Smart Grid Data Integrity Attacks : Characterizations and Countermeasures π," no. 025478, pp. 232–237, 2011.

[50] H. Sandberg, "Stealth Attacks and Protection Schemes for State Estimators in Power Systems," pp. 214–219, 2010.

[51] F. Pasqualetti, D. Florian, and F. Bullo, "Cyber-Physical Attacks in Power Networks : Models , Fundamental Limitations and Monitor Design," pp. 2195–2201, 2011.

[52] A. Teixeira, S. Amin, H. Sandberg, K. H. Johansson, and S. S. Sastry, "Cyber security analysis of state estimators in electric power systems," *Proc. 49th IEEE Conf. Decis. Control*, pp. 5991–5998, 2010.

[53] H. Sandberg and H. Johansson, "A Cyber Security Study of a SCADA Energy Management System : Stealthy Deception Attacks on the State Estimator," no. August, pp. 1–11, 2010.

[54] N. M. Manousakis and G. N. Korres, "Optimal PMU Placement for Numerical Observability Considering Fixed Channel Capacity-A Semidefinite Programming Approach," *IEEE Trans. Power Syst.*, vol. 31, no. 4, pp. 3328–3329, 2016.

[55] T. T. Kim and H. V. Poor, "Strategic Protection Against Data Injection Attacks on Power Grids," vol. 2, no. 2, pp. 326–333, 2011.

[56] S. Bi, Y. Jun, A. Zhang, and S. Member, “Graphical Methods for Defense Against False-Data Injection Attacks on Power System State Estimation,” vol. 5, no. 3, pp. 1216–1227, 2014.

[57] “NERC-CIP, Critical Infrastructure Protection,” *North American Electric Reliability Corporation*. [Online]. Available: http://www.nerc.com/pa/Stand/Pages/CIPStandards.aspx. [Accessed: 02-Sep-2018].

[58] “Mandatory Reliability Standards for Critical Infrastructure Protection,” *Fed. Energy Regul. Commision*, vol. 215, no. 706, 2008.

[59] North American Electric Reliability Council (NERC), “Critical Infrastructure Protection Standards,” 2014. [Online]. Available: http://www.nerc.com/pa/Stand/Pages/CIPStandards.aspx.

[60] AGA, “Cryptographic Protection of SCADA Communications. Part 1: Background, Policies and Test Plan,” *Am. Gas*, no. 12, pp. 1–123, 2006.

[61] J. Eisenhauer, P. Donnelly, M. Ellis, and M. O’Brien, “Roadmap to Secure Control Systems in the Energy Sector,” 2006.

[62] “National Supervisory Control and Data Acquisition (SCADA).” [Online]. Available: http://energy.sandia.gov/energy/ssrei/gridmod/cyber-security-for-electric-infrastructure/scada-systems/.

[63] “No Title.” [Online]. Available: https://www.hartcomm2.org/frontpage/wirelesshart.html.

[64] “SCADA Hacking: The Key Differences between Security of SCADA and Traditional IT systems.” .

[65] D. Dolezilek, K. Carson, K. Leech, K. Streett, and ., “Secure SCADA and Engineering Access Communications: A Case Study of Private and Public Communication Link Security,” *Schweitzer Eng. Lab.*, p. , 2006.

[66] J. Abshier, “10 Principles for securing control systems,” *Control*, vol. 18, no. 10, pp. 77–81, 2005.

[67] J. D. Fernandez and A. E. Fernandez, “SCADA systems: Vulnerabilities and remediation,” *J. Comput. Sci. Coll.*, vol. 20, pp. 160–168, 2005.

[68] D. Geer, “Security of critical control systems sparks concern,” *Computer (Long. Beach. Calif).*, vol. 39, no. 1, pp. 20–23, 2006.

[69] M. Munir, F. Parisi-Presicce, and W. Duminda, “DNPsec: A Secure Framework for DNP3

in SCADA systemsDNPsec: A Secure Framework for DNP3 in SCADA systems," in *International Joint Conference on Computer Information and Systems Sciences and Engineering*, 2005.

[70] M. . McDonald, G. . Conrad, T. C. Service, and R. . Cassidy, "Cyber Effects Analysis using VCSE," Albuquerque, NM, USA.

[71] D. C. Bergman, D. Jin, D. M. Nicol, and T. Yardley, "The virtual power system testbed and inter-testbed integration," *Proc. 2nd Conf. Cyber Secur. Exp. test*, no. August, p. 5, 2009.

[72] V. Salehi, A. Mohamed, A. Mazloomzadeh, and O. A. Mohammed, "Laboratory-based smart power system, part I: Design and system development," *IEEE Trans. Smart Grid*, vol. 3, no. 3, pp. 1394–1404, 2012.

[73] V. Salehi, A. Mohamed, A. Mazloomzadeh, and O. A. Mohammed, "Laboratory-based smart power system, part II: Control, monitoring, and protection," *IEEE Trans. Smart Grid*, vol. 3, no. 3, pp. 1405–1417, 2012.

[74] M. Mallouhi, Y. Al-Nashif, D. Cox, T. Chadaga, and S. Hariri, "A testbed for analyzing security of SCADA control systems (TASSCS)," *IEEE PES Innov. Smart Grid Technol. Conf. Eur. ISGT Eur.*, pp. 1–7, 2011.

[75] J. Hong *et al.*, "An intrusion and defense testbed in a cyber-power system environment," *IEEE Power Energy Soc. Gen. Meet.*, pp. 1–5, 2011.

[76] C. Queiroz, A. Mahmood, and Z. Tari, "SCADASimA framework for building SCADA simulations," *IEEE Trans. Smart Grid*, vol. 2, no. 4, pp. 589–597, 2011.

[77] M. Thomas and E. Al, "A CONTROL SYSTEM TESTBED TO VALIDATE CRITICAL INFRASTRUCTURE PROTECTION CONCEPTS," *Ceps*, no. December, pp. 12–13, 2011.

[78] J. Mirkovic and T. Benzel, "Teaching cybersecurity with DeterLab," *IEEE Secur. Priv.*, vol. 10, no. 1, pp. 73–76, 2012.

[79] A. Hahn, A. Ashok, S. Sridhar, and M. Govindarasu, "Cyber-physical security testbeds: Architecture, application, and evaluation for smart grid," *IEEE Trans. Smart Grid*, vol. 4, no. 2, pp. 847–855, 2013.

[80] B. Chen, K. L. Butler-Purry, A. Goulart, and D. Kundur, "Implementing a real-time cyber-physical system test bed in RTDS and OPNET," *2014 North Am. Power Symp. NAPS*

2014, pp. 1–6, 2014.

[81] B. Chen, N. Pattanaik, A. Goulart, K. L. Butler-Purry, and D. Kundur, “Implementing attacks for modbus/TCP protocol in a real-time cyber physical system test bed,” *Proc. - CQR 2015 2015 IEEE Int. Work. Tech. Comm. Commun. Qual. Reliab.*, 2015.

[82] C. B. Vellaithurai, S. S. Biswas, and A. K. Srivastava, “Development and Application of a Real-Time Test Bed for Cyber–Physical System,” *IEEE Syst. J.*, vol. 11, no. 4, pp. 1–12, 2015.

[83] S. S. Biswas, F. Shariatzadeh, R. Beckstrom, and A. K. Srivastava, “Real time testing and validation of Smart Grid devices and algorithms,” *IEEE Power Energy Soc. Gen. Meet.*, 2013.

[84] B. A. Vaccaro, M. Popov, D. Villacci, and V. Terzija, “An Integrated Framework for Smart Microgrids Modeling , Communication , and Verification,” *Proc. IEEE*, vol. 99, no. 1, pp. 119–132, 2011.

[85] G. Koutsandria, R. Gentz, M. Jamei, A. Scaglione, S. Peisert, and C. McParland, “A Real-Time Testbed Environment for Cyber-Physical Security on the Power Grid,” *Proc. First ACM Work. Cyber-Physical Syst. and/or Priv. - CPS-SPC ’15*, pp. 67–78, 2015.

[86] “Test Bed for a Cyber-Physical System Based on Integration of Advanced Power Laboratory and eXtensible Messaging.” [Online]. Available: https://www.ece.cmu.edu/~electriconf/posterpdf_2015/Matin_Meskin%0APoster.pdf.

[87] Y. Yang *et al.*, “Cybersecurity test-bed for IEC 61850 based smart substations,” *IEEE Power Energy Soc. Gen. Meet.*, vol. 2015–Septe, pp. 1–5, 2015.

[88] “INEL Test Range, Protecting Nation’s Infrastructure.” [Online]. Available: http://www4vip.inl.gov/research/%0Aidaho-test-range/d/idaho-test-range.pdf.

[89] N. Saxena, V. Chukwuka, L. Xiong, and S. Grijalva, “CPSA : A Cyber-Physical Security Assessment Tool for Situational Awareness in Smart Grid,” pp. 69–79, 2017.

[90] B. Y. Mo *et al.*, “Cyber – Physical Security of a Smart Grid Infrastructure,” *Proc. IEEE*, vol. 100, no. 1, pp. 195–209, 2012.

[91] C. W. Ten, G. Manimaran, and C. C. Liu, “Cybersecurity for critical infrastructures: Attack and defense modeling,” *IEEE Trans. Syst. Man, Cybern. Part ASystems Humans*, vol. 40, no. 4, pp. 853–865, 2010.

[92] Y. Yang *et al.*, “Man-in-the-middle attack test-bed investigating cyber-security

vulnerabilities in smart grid SCADA systems," *Int. Conf. Sustain. Power Gener. Supply (SUPERGEN 2012)*, no. July, pp. 138–138, 2012.

[93] J. Wei and D. Kundur, "A flocking-based model for DoS-resilient communication routing in smart grid," *GLOBECOM - IEEE Glob. Telecommun. Conf.*, pp. 3519–3524, 2012.

[94] T. T. Tran, O. S. Shin, and J. H. Lee, "Detection of replay attacks in smart grid systems," *2013 Int. Conf. Comput. Manag. Telecommun. ComManTel 2013*, pp. 298–302, 2013.

[95] P. Y. Chen, S. M. Cheng, and K. C. Chen, "Smart attacks in smart grid communication networks," *IEEE Commun. Mag.*, vol. 50, no. 8, pp. 24–29, 2012.

[96] A. Abur and G. Exposito, *Power System State Estimation: Theory and Implementation*. New York, NY: Marcel Dekker, 2004.

[97] P. Kundur, N. . Balu, and M. . Lauby, *Power System Stability and Control*, Vol 4, . New York: Mcgraw-hill, 1994.

[98] Y. Sun and T. J. Overbye, "Visualizations for power system contingency analysis data," *IEEE Trans. Power Syst.*, vol. 19, no. 4, pp. 1859–1866, 2004.

[99] C. Vellaithurai, A. Srivastava, S. Zonouz, and R. Berthier, "CPIndex: Cyber-physical vulnerability assessment for power-grid infrastructures," *IEEE Trans. Smart Grid*, vol. 6, no. 2, pp. 566–575, 2015.

[100] S. Zonouz, A. Houmansadr, and P. Haghani, "EliMet: Security metric elicitation in power grid critical infrastructures by observing system administrators' responsive behavior," *Proc. Int. Conf. Dependable Syst. Networks*, 2012.

[101] "Suricata - Open source network threat detection engine." [Online]. Available: https://suricata-ids.org.

[102] "Interoperability and cyber security plan, nreca crn smart grid regional demonstration, grant de-oe-0000222."

[103] N. Saxena and S. Grijalva, "Dynamic Secrets and Secret Keys Based Scheme for Securing Last Mile Smart Grid Wireless Communication," *IEEE Trans. Ind. Informatics*, vol. 13, no. 3, pp. 1482–1491, 2017.

[104] R. Kinney, P. Crucitti, R. Albert, and V. Latora, "Modeling cascading failures in the North American power grid," *Eur. Phys. J. B*, vol. 46, no. 1, pp. 101–107, 2005.

[105] M. Vaiman *et al.*, "Risk assessment of cascading outages: Methodologies and challenges," *IEEE Trans. Power Syst.*, vol. 27, no. 2, pp. 631–641, 2012.

[106] S. Arianos, E. Bompard, A. Carbone, and F. Xue, “Power grids vulnerability: a complex network approach,” pp. 1–16, 2008.

[107] T. Liu, Y. Gu, D. Wang, Y. Gui, and X. Guan, “A novel method to detect bad data injection attack in smart grid,” *2013 Proc. IEEE INFOCOM*, pp. 3423–3428, 2013.

[108] A. Ashok, M. Govindarasu, and V. Ajjarapu, “Online Detection of Stealthy False Data Injection Attacks in Power System State Estimation,” *IEEE Trans. Smart Grid*, vol. 3053, no. c, pp. 1–1, 2016.

[109] S. Li, Y. Yilmaz, and X. Wang, “Quickest Detection of False Data Injection Attack in Wide-Area Smart Grids,” *IEEE Trans. Smart Grid*, vol. 6, no. 6, pp. 2725–2735, 2015.

[110] L. Wang, T. Islam, T. Long, A. Singhal, and S. Jajodia, “An attack graph-based probabilistic security metric,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 5094 LNCS, pp. 283–296, 2008.

[111] T. Base and T. Base, “Common Vulnerability Scoring System v3 . 0 Examples,” no. July, pp. 1–38, 2016.

[112] J. Davis and S. Magrath, “A Survey of Cyber Ranges and Testbeds,” p. 29, 2013.

[113] R. Buyya and M. Murshed, “GridSim: a toolkit for the modeling and simulation of distributed resource management and scheduling for Grid computing,” *Concurr. Comput. Pract. Exp.*, vol. 14, no. 13–15, pp. 1175–1220, 2002.

[114] R. Roche, S. Natarajan, A. Bhattacharyya, and S. Suryanarayanan, “A framework for co-simulation of AI tools with power systems analysis software,” *Proc. - Int. Work. Database Expert Syst. Appl. DEXA*, pp. 350–354, 2012.